MAY/JUNE '87
Vol. 3 No. 4
$3.00 U.S. Funds

AMERICA'S LARGEST TIMEX SINCLAIR MAGAZINE

TIME DESIGN

MAGAZINE

Learn
Stan Lemke's
Method of
ENTERING PROGRAM
Listings
without
Headaches!
Try
CK — TYPE
For Your TS2068

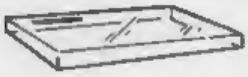
Complete
MIDWEST
TS Computer Fest
COVERAGE

ABACUS
SPREADSHEET
"TIPS"
For Your QL

Turn that meek
ZX81
into a
MONSTER...
with
Tim Stoddard's
INTERNAL
64K RAM
MODIFICATION

COMPUTER DUST COVERS

NYLON DUST COVERS — COMPLETE WITH **TIMEX/SINCLAIR** LOGO!



T/S2068 (GREY)

T/S1000 (BLUE)

BUY ANY TWO AND SAVE 10%

893-821	T/S2068 DUSTCOVER	$14.95
893-822	T/S1000 DUSTCOVER	7.95
893-823	T/S1500 DUSTCOVER	7.95
893-824	T/S2040 DUSTCOVER	10.95
893-825	QL DUSTCOVER (WITH QL LOGO)	17.95

T/S2040 (BLUE)

T/S1500 (GREY)

BLANK DATA TAPES

		5 lot ea.	10 lot ea.	25 lot ea.	100 lot ea.
817-001	C-5	.75	.70	.60	.50
817-002	C-10	.80	.75	.65	.55
817-003	C-12	.85	.80	.70	.60
817-004	C-20	.85	.80	.70	.60
817-005	C-30	.90	.85	.75	.65

TAPES CAN BE MIXED FOR QUANTITY PRICING
(MINIMUM 25 pieces)

PLEASE SPECIFY WHITE OR BLACK CASING.

TRACTOR FEED

Cassette Labels

	10 PACK	100 PACK
817-501W WHITE	.40	2.95
817-501R RED	.50	3.95
817-501B BLUE	.50	3.95
817-501G GREEN	.50	3.95
817-501Y YELLOW	.50	3.95
817-501L LAVENDER	.50	3.95

STACKABLE CASSETTE CASES

A SUNSET ELECTRONICS EXCLUSIVE!

SPRING—LOADED OPENING!
STACK VERTICALLY or HORIZONTALLY!

		5 lot	10 lot	25 lot	100 lot
878-001	CASE _____ ea.	.50	.45	.40	.30

FRONT LABELS INCLUDED

F R E E !

These new style cases are available at no extra charge when purchasing programs. Just specify which type of case you prefer —

STACKABLE, NORELCO or **POLYSOFT**

ORIGINALLY SOLD FOR $1.00 EACH.
ADD $3.00 SHIPPING TO ALL ORDERS

PHONE:
ORDERS ONLY:
(415) 665-6161

INFORMATION
(415) 665-8330

SUNSET ELECTRONICS

2254 Taraval Street
San Francisco, CA 94116

HOURS:
M - F 10-6
SAT 10-5

The Editor's Forum

Thanks to Tim Stoddard, I had the chance to "play" with a ZX81 the past two months. But, this was no ordinary ZX. Tim generously loaned me his "souped-up" ZX81 to demo at the TS Computer Fest. This computer contained the internal 64k RAM modification that is described in his issue; plus attached to the rear buss, was his proto-type D.A.M. board (Data Acquisition Module). If you remember (see Jan/Feb '87 and March/April '87 issues of TDM) the board contains an analog-to digital converter, a digital-to-analog converter, and a real time clock. With the software that Tim had written, our demo could measure a room temperature reading from a small probe mounted at the top of the D.A.M. board, and display the reading on the screen. Also, a voltage in the range of 0 to 2.55 could be selected and typed in at the keyboard. The selected value would be printed on the screen and could be measured with a meter at two pins on the D.A.M. board. Not to mention that the real-time clock continuosly displayed the date and time. All of this operated at the same time on just one little 'ol ZX81.

However, the most amazing thing was that Tim's computer didn't behave like the ZX81 I used to have. Let me explain. When Stephanie and I arrived at the Holiday Inn on Friday, the day before the show, we had a chance to set up early. I went ahead and put together the computer demo to make sure everything was going to work properly. Accidently, the ZX was left plugged-in overnite. The next morning (the day of the show) when I discovered it, the computer was just barely warm. It "fired up" up ok...no problems. During that day, the guests arrived and literally crammed into the exhibit room. With all of the computers and monitors (and warm bodies), the room temperature soared to 85°F (until some kind soul fixed the air conditioner). The ZX81 never once overheated or crashed it's program during the entire Fest! Something could be said of Tim's efficient internal 64k RAM design and the proper heat-sinking he used.

Along with the RAM upgrade article, we have news and photographs from the 1987 Midwest TS Computer Fest. Attending the Fest has made this issue come out a bit late. For this I apologize...but we should be back on track next issue. Also, a few of our features have been postponed until the next issue. For example Paul Bingham was to have the second installment of his CLASSY FRONT END series published in this issue. But I gave Paul some time off for a very good reason--he had a new addition to his family. A new son, Spencer Christian Bingham, was born on April 21st. Congratulations to a proud Dad and Mom. We'll see Paul back again next issue.

In closing, I might add that if there is something special you would like to see printed in an upcoming issue (perhaps an article or program for your computer in an area of special interest), just drop me a short note, and I'll see what I can do. Until then, "Happy Computing" and "don't give up on the chip!"

Sincerely,
Tim Woods
Managing Editor
Time Designs Magazine Co.

Special Information for TDM Subscribers

WHEN TO RENEW? To determine your expiration date, read the information in the upper-right corner of your shipping label (front cover). As an example: "May/87" means that the May/June 1987 issue will be your last one until you renew. If you forget, we will send you a reminder notice, but an early renewal is appreciated. You can use the form on page 38 if you like.

PRICE INCREASE IS COMING: We have held the price of an annual subscription at $15/year for the past three years, while improving quality and increasing page count. Now our paper supplier has informed us of a 15% price hike, and the Postal Service will be raising their rates this coming April. Therefore, we are forced to increase our subscription rate to $16.95 starting July 20th. (Our only competitor starting charging $16.95 a few years ago, and their page count is much less than TDM's.) All subscriptions and renewals that we receive before July 20th, will be guaranteed the old rate of $15/year. After that--$16.95. Our promise to you: to continue publishing a good Timex/Sinclair magazine.

PLANNING ON MOVING? Please let us know of any changes as soon as possible. The post office will not reliably forward second class mail (like this magazine). To ensure no issues will be lost--let us know your new address right away. We are not responsible for lost magazines for this reason.

TIME DESIGNS MAGAZINE

FOR ALL TIMEX AND SINCLAIR COMPUTERS

TIME DESIGNS MAGAZINE CO.
29722 Hult Rd. • Colton, Oregon 97017

(503) 824-2658

CompuServe ID# 71350,3230

MAY/JUNE '87
Vol. 3 No. 4

Editor: Tim Woods

Assistant Editor: Stephanie Woods

Editorial Assistant/Production: D.L. Woods

Photography:
(unless otherwise noted): Thomas Judd
Printing by: Toad'l Litho Printing and Comp., Oregon City, Oregon 97045

SUBSCRIPTIONS: $15 a year for six issues (US funds only). No extra charge to Canadian subscribers. All other countries please write for information on air mail rates.

CUSTOMER SERVICE: Customer satisfaction is our goal. For subscription service problems please write or call TIME DESIGNS.

CHANGE OF ADDRESS: Write or call to prevent delay of sevice.

WMJ DATA SYSTEMS

Serving the Timex/Sinclair community since 1983

- Free shipping on all prepaid orders!
- Shipment within 24 hours on most of our orders!
- Your personal check is gladly accepted with no delay!
- C.O.D. phone orders accepted! (Shipping charges added)

TS1000

ACZ GENERAL LEDGER	TS1000	$19.95
SUPERTAPE-45 PROGRAMS	TS1000	18.95
TEXTWRITER 1000-32 col.	TS1000	12.95
SPEECH SYNTHESIZER	TS1000	16.95
STOCK PLOT	TS1000	14.00

BOOKS AND ACCESSORIES

KRAFT SYSTEM JOYSTICKS	TS2068	$ 9.95
HACKER'S HANDBOOK	ALL	9.95
BASICS AND BEYOND	TS2068	7.00
C10 CASSETTES	25 for	21.00
C20 CASSETTES	25 for	23.00

TS2068

ACZ GENERAL LEDGER	TS2068	$19.95
THE DEALER'S DEN	TS2068	19.95
MACHINE CODE TUTOR	TS2068	18.50
TOMAHAWK	TS2068	16.95
NIGHT GUNNER	TS2068	16.95
FIGHTER PILOT	TS2068	16.95
CRITICAL MASS	TS2068	15.95
SABOTEUR	TS2068	15.95
BASIC TOOLKIT	TS2068	13.95
MSCRIPT (specify conf.)	TS2068	24.95
DIAMOND MIKE	TS2068	16.95
COLONIZE THE UNIVERSE	TS2068	16.95
STOCK PLOT	TS2068	16.00
All QUICKSILVA Software	TS2068	CALL

WMJ DATA SYSTEMS—TD

4 Butterfly Drive
Hauppauge, NY 11788

(516) 543-5252 (2 p.m. - 5 p.m. Best)

FREE Apple, Atari, Commodore, IBM...Catalog
FREE Timex/Sinclair Catalog - Write or Give us a CALL!

LETTERS

TS2068 DISK DRIVE SYSTEMS

Dear Tim,

I have several questions/comments which may be of interest to other readers of TIME DESIGNS. I don't think I've seen any comprehensive article on the various disk systems that are available for the Timex Sinclair 2068. I have seen isolated advertiser's references and offerings on several of these, but I don't really know which way to jump to get the best deal of these for my purposes.

I have the Russell Speech Synthesizer for the ZX81. I've seen no reference on adaptation of this to the TS2068. I also have the TREE System FORTH ROM for the ZX81; I've seen no adaptation or anything similar for the TS2068.

I am still enjoying your magazine, as the best and now almost only source of Sinclair/Timex info and news.

Sincerely,
Louis G. Dooley
Ocala, FL

Editor: Good questions. I'll tackle them in reverse order. First, the best source of FORTH language information for Sinclair computers in the U.S., is to contact Gary Ganger of the Dayton (Ohio) Microcomputer Association. He is the founder of the FORTH Interest Group (FIG), a sub-group of DMA. Gary is very familiar with versions of the language for the ZX81/2068 and Spectrum, and is willing to help. You can contact him by writing to: 812 Hedwick St., New Carlisle, Ohio, 45344. As for your second question, I'm sure it is possible, but haven't seen or heard of anyone adapting it to the TS2068. Have you tried contacting Russell Electronics? Perhaps one of our readers have some information. Finally, the topic of disk drives. Indeed we are guilty of not providing any major follow up to articles we have published in the past of available disk drive systems for the TS2068. I hope to correct that very soon. In the meantime, here's a quick overview on what's out there. A total of five different systems have been released for the standard (non-Spectrumized) 2068. Two of the systems are no longer available (Ramex Millenia K and Zebra's FDD-3000), which leaves the 2068 user with three to choose from. They are the Aerco FD-68 (Box 18093, Austin, Texas 78760), the Larkin Disk Interface (RR#2, Navan, Ont., Canada, K4B-1H9), and the Oliger Disk Drive Interface (11601 Whidbey Dr., Cumberland, IN 46229). It should be understood that the three models come as an interface circuit card and you must provide the drives/case/power supply for the drives (Aerco does offer an optional disk drive package along with their interface). Your best bet on drives would be to pick up a copy of COMPUTER SHOPPER, or consult your local TS dealer. (I know of one, RMG Enterprises, that stocks drives). It would be wise to write for information and prices on the three systems, in order to compare features, etc. Some of them offer "extras", such as an RGB monitor interface, or Spectrum compatibility, and NMI "magic button" SAVEs. If you have Spectrum emulated your 2068 and wouldn't mind a genuine Spectrum disk drive system, you might want to check out the new DISCIPLE Interface (Rockfort Products, 81 Church Road, London, England, NW4 4DP) which sells for £89.95 in the U.K. Along with a disk interface, the Disciple also has a built-in Centronics printer interface, joystick ports, networking capabilities, and more. Quite an impressive unit that looks a lot like the Sinclair Interface One. I think we will be seeing more of this one in the months to come. Other Spectrum disk drive interfaces are available, such as the OPUS Discovery. You were right by stating you want a disk drive system that would fit your particular needs. That's the key to selection...and be sure to shop around, and (if possible) ask questions of other users who have such systems.

QL NEWS FROM THE U.K.

Dear Mr. Woods,

From this side of the pond a number of QL facts have come to light:

1. Digital Precision is about to launch it's own desktop publishing package. 2. All those who paid in advance for a "FUTURA" machine have had their money re-funded because, though advertised months ago, it appears not to have reached the circuit board stage. 3. The "THOR 20" complete with 68020 and an optional 68881 fp co-processor is at the circuit board stage and expected to be seen in mid-May (time will tell). Projected prices are £425 (without 68881) and £600 (with) for a 12.7 MHz version. 16.7 MHz versions will be twice this. 4. QRAM Utilities, the front end for the FUTURA (as was/is/maybe), appears to have been written not to work with programs compiled using the DP Supercharge and Turbo SuperBASIC compilers due to bad feelings between the companies. 5. A new adventure writing system with real-time simulation surfaced recently. 6. It is rumoured that the QL's multi-tasking QDOS operating system is not actually owned by anybody, unlike SuperBASIC. Several companies are playing safe though, and are producing new compatible systems. Complete QDOS disassemblies are available from a number of companies including Sandy.

All the best with TDM.

Yours Sincerely,
Richard Howe
Proprietor
ARK DISTRIBUTION
Isle of Wight, United Kingdom

LOAD LOADER

BY
WILLIAM C. ANDREWS

To dress up the monitor screen while loading a program I use the following short program. LINE 4 is the title to be displayed (up to 20 CHR$). LINE 9 is the actual program name ---limit 10 CHR$ (7 with A & J after the $). LINE 10 for a machine code (if needed). LINE 11 is the loading title of this program. Edit LINE 11, delete the line number and REM and save onto tape or wafer before the main program.

```
   1 BORDER 1: PAPER 1: CLS
   2 LET L$="______________________
________________"
   3 LET K$="______________________
________________"
   4 LET N$="??"
   5 PRINT AT 4,(31-(LEN N$+6))/
2; PAPER 2; INK 0; BRIGHT 1;"■";
K$( TO LEN N$+4);"■"
   6 PRINT AT 5,(31-(LEN N$+6))/
2; PAPER 2; INK 0; BRIGHT 1;"■ "
; INK 7;"$ ";N$;" $"; I
NK 0;" ■"
   7 PRINT AT 6,(31-(LEN N$+6))/
2; PAPER 2; INK 0; BRIGHT 1;"■";
L$( TO LEN N$+4);"■"
   8 PRINT AT 10,11; PAPER 6; IN
K 9; FLASH 1;" LOADING "; FLASH
0; PAPER 0;AT 15,9;" PLEASE WAIT
 "
   9 INK 1: LOAD "??"
  10 LOAD "??"CODE
  11 REM  SAVE "??" LINE 1
```


More Letters Next Page...

L E T T E R S

MILLENNIA K

Where are the MILLENNIA K users? Surely most of you out there remember seeing an ad or two about a disk drive system for the TS2068 that bragged about the disk operating system that would allow 1,000,000 byte storage on a single disk. This letter is directed to the ones of you who had the nerve to fork out the $450 or so required to obtain this reputably fabulous disk system from the now defunct RAMEX.

The brag turned out to be fact! The system turned out to be one of the easiest disk systems, to date, to use. It uses all the tape commands and the "extra" commands like FORMAT, CAT, ERASE, MOVE, etc. Not only that, it allows the use of "sequential files" which few Sinclair users have had the opportunity to use. However, you already know that because you bought one, right?

If you are reading this letter, the publishers of TIME DESIGNS have seen fit to open the new arena of SPDOS information. The desire is to provide support to some new orphans in the form of exploration of the interface and the supporting SPDOS. If there are any questions about methods of implementing programs using sequential files or expanding the command list, or getting programs to work with the system, just write in and I'll do my best to answer the query or find the answer from some of you.

Please Write To:
Munson H. Cockayne, Jr
342 Trotter Court
Sanford, Florida 32771

QL USERS PLEA: "TO BE OR NOT TO BE?"

HACKERS ALERT. The Bard hit the QL nail right on the head when he asked, "to be or not to be, that is the question". Fairly put, the question--is the QL a capable User "can do" computer, or are we "hacking" it to death?

If the QL is such a good programer/hacker's machine, then how come there aren't any "made in the USA" programs? We need American programs designed for the American non-programmer User's use! This includes a "load and useable" database program, comprehensive accounting and bookkeeping programs for business, non-profit church and home use, and a lot more. The only difference between a programer/hacker and the non-programer User, is that the hacker is a "self sufficient" user.

If, the knowledgeable hackers would stop their "self sufficiency" long enough to write some of these needed programs, it would re-vitalize the QL's perception. By so doing, they could give our QL the "Dual Life Expectancy" that it deserves. So hackers, ask not what the QL can do for YOU, rather what YOU can do for the QL! If YOU don't, who will? Do it before it is too late.

Kenton Garrett
Lansing, KS

Editor: Your points are well taken. But let me raise an issue which you may not have thought of. The actual numbers of QL's currently in use here in the U.S. (not to mention the number of bonafide hacker/programer's) is very low. I have heard that estimates range between 2 to 5 thousand units. That factor may have something to do with the lack of user-oriented "serious" software. Yet in many European countries (i.e., France, Germany, etc.), ten times more QL's were sold...and explains why they are releasing a substantial amount of both hardware and software support. Perhaps your letter will strike some chords and stir some interest. I couldn't agree with you more about the need for programs that are geared to the user rather than the engineer-types, or require one to digest documentation that suspiciously looks like it was written in Latin.

HURRAH FOR NOVELSOFT

Recently I purchased TIMACHINE from Novelsoft. Try as I might, the program wouldn't LOAD; it was the first tape I have ever been unable to LOAD or SAVE immediately. I wrote Novelsoft detailing the steps I had taken, along with some print-outs of various segments I was able to access, asking for their help in resolving the problem.

Shortly after, I received from David Ridge, what he believed to be the answer to the problem, along with a request that should I have any further difficulty, to contact him for further assistance. Everything worked fine.

I thought it would be nice, were you to include some mention of this in your column. It is very reassuring that TS users can deal with people as responsible as Novelsoft, even in a market as limited as the Timex Sinclair.

Yours truly,
Francis C. Dupre
Southampton, PA

APRIL FOOLER

Well, I guess it's time to renew my subscription and it's my pleasure. I look forward to each issue of TDM and I thank you for keeping the faith in the TS computer systems.

I was going to write about the modem program in your last issue, about how I interfaced the simple modem to my 2068 and how I ran the program and about how I connected with dozens of mail boxes. I won't do that but I hope next year somebody comes up with a fantastic APRIL FOOLS gag for the author of that article.

Warren Tucker
Vallejo, CA

NEW CAR SHOPPER

Dear Tim,

Here is a little program I wrote to compare car prices and figure approximate loan payments. It tells me I can't afford to buy the cars I already own!

Paul Hill
SINCUS NEWS
Johnson City, NY

```
   5 REM data from CONSUMER REPO
RT Magazine
  10 INPUT "Car Name ";n$;" Mode
l ";m$
  20 PRINT n$: PRINT m$: INPUT "
Sticker Price ";s
  30 PRINT "Sticker Price $";s:
INPUT "Cost Factor ";c: LET c=c/
100: LET Deal=cos
  40 PRINT "Dealer Cost $";Deal
  50 INPUT "Destination Costs ";
des: LET Cost=deal+des
  60 PRINT "Plus Shipping Costs
";des;" = $";Cost
  70 INPUT "Total Options Sticke
r Price ";opt;" Dealer Cost Fact
or ";def: LET OPTS=opt*def: LET
Cost=Cost+OPTS
  80 PRINT "Dealer Cost For Car
= $";Cost: LET stick=s+opt: REM
check sticker price for destinat
ion charges.
  90 PRINT "Barginning Range Is
";Cost;" to ";stick: LET def=sti
ck-Cost: LET loan=Cost+(.5*def)
 100 PRINT "Approx. Buying Price
= $";loan
 110 INPUT "Downpayment/Trade-In
Value ";dow: LET loan=loan-dow
 120 LET loan=loan+(sales tax if
 applicable)
 125 REM delete Line 120 if no s
ales tax figure applies, otherwi
se program will not work properl
y
 130 LET r1=.19: LET r2=.205: LE
T r3=.215: REM approx a 9% rate,
 adjust r values as necessary.
 140 LET loan1=(loan+(r1*loan))/
36: PRINT "3 Year Loan Payment =
$";loan1
 150 LET loan2=(loan+(r2*loan))/
48: PRINT "4 Year Loan Payment =
$";loan2
 160 LET loan3=(loan+(r3*loan))/
60: PRINT "5 Year Loan Payment =
$";loan3
 170 REM add printouts to carry
around with you as you shop!
```

For Your Sinclair

New Releases

Last issue we reported on an external keyboard interface that uses the cartridge port of the TS2068, available from John Mathewson (1852 Appleford St., Gloucester, Ontario, Canada K1J 6T4). John has been busy designing new peripherals for the TS2068, and now has several other new items. The "Sound Booster" plugs into the rear port and amplifies both the BEEP and SOUND information sufficient to drive an 8 ohm, 8" speaker, or there is a low level output provided to drive another power amp or stereo amp. The on-board 9 volt battery helps to reserve the computer's internal power for other devices. A feed-thru buss is also included. The "Sound Booster" board is priced at $41.50 U.S. funds (battery and speaker are not included). An RGB interface is now available for $34.95 U.S. funds, and provides an output to drive any RGB-type color monitors. Press-on type Key Caps for external keyboards are available for $5 a set. A "Cartridge Adapter" card has also been designed to plug any cartridge into the rear buss of the TS2068 whenever the cartridge dock is occupied. Write for further information.

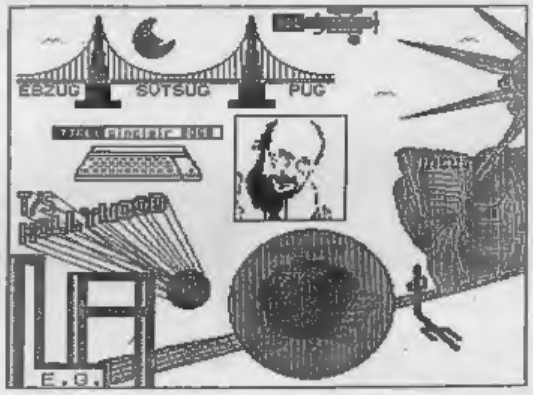
A new machine code utility software package is available for the ZX81/TS1000/TS1500 called KAPKIT 1000. A number of special routines are included that will save the ZX/TS programmer time and allow greater flexibility. Move whole programs or variables to high or low RAM and back, convert hex to decimal, delete more than one program line at a time, and much more. A cassette tape and complete documentation are available from: LST Software, Box 62, Alcester, SD 57001, for $14.95 plus $1.95 for S&H. The program is also available from E.Arthur Brown.

You may have noticed that Commodore's inexpensive 1520 Printer/Plotter has dropped in price. One of our long time readers (and occaisonal contributor), John McMichael, has devised an interface and companion software driver to operate the 1520 with a Timex Sinclair 2068. The result is high-resolution four color graphic plotting with the Timex, using simple LPRINT commands. For complete details and prices, send a legal SASE to: John McMichael, 1710 Palmer Dr., Laramie, WY 82070. He will even provide you with an address where you can buy the plotter for $49.95.

PC-DRAW is a software package just released by a promising new company called MDM Enterprises. The impressive new program allows the user to design detailed printed circuit board artwork, which can then be printed and photographed, providing a negative for circuit board etching. Circuit drawing is made easy with joystick control, and the user-friendly documentation takes you through the procedures step by step. PC-DRAW supports all of the popular TS printer interfaces and is set up for Epson compatible printers. Similar programs for other computers are expensive. This one sells for $19.95 plus $3.00 (total order) S&H and is exclusively available from: Knighted Computers, 707 Highland St., Fulton, NY 13069, tel. (315) 593-8219.

CompuServe Information Service has introduced a new graphics medium, that will eventually replace the RLE graphics (for information on this subject, reference Stan Lemke's article on RLE in the Jan/Feb '87 issue of TDM). The new standard is called GIF (for Graphics Interchange Format). With the appropriate software, a picture file could be "downloaded" via a modem from CompuServe, then displayed on the screen or sent to a printer. GIF allows for full color and high resolution on many types of personal computers. Larry Wood of the Picture Forum (GO PICS) on CompuServe, recently told TDM that information would be released to Sinclair programmers to see if a GIF decoder or encoder program is feasible with our computers.

Speaking of telecommunications, the second edition of "The Guide To T/S Telecommunications" by Pete Fischer and Steve Ishii is out...and is it ever a real gem! The new deluxe and expanded edition hardly resembles it's forerunner. The guide was re-printed using a laser printer and is much more readable. The front cover is actually an RLE graphic that was passed around to a number of TS users who added their own design, resulting in an interesting collage of pictures from around the U.S. Well worth the $5 price. Get your copy from: Pete Fischer, P.O. Box 2002, Tempe, AZ 85281. Pete showed his second edition guide for the first time, at the Midwest TS Computer Fest in Indianapolis.



This RLE graphic was specially produced for the cover of the new deluxe edition of "The Guide To T/S Telecommunications". Several TS users from around the U.S. took part in drawing the graphics that make up the picture.

Having trouble figuring out the new tax laws that were recently passed by our U.S Congress? A special 2068 program just might be your ticket. Herb Bowers, a former Federal Auditor and private tax preparer has put together a comprehensive tax software package that contains two separate programs. "THE NEW TAX LAW AND YOU" is priced at $12.00 postage paid, and is available from the author: Herb Bowers, Sr., 2588 Woodshire Circle, Chesapeake, VA 23323, tel. (804) 487-5924. Be a tax "expert" and impress your friends with the astonishing accuracy that your TS2068 can calculate the new tax code.

Charles Stelding has a TS2068 desktop publishing software package, and now has released a version just for Olivetti 2300 Ink-Jet printer owners (The WJDJUP Company's "Winkjet 1" printing utility is required). For a sample print-out and demonstration of what the program can do, send a legal SASE to obtain information and prices to: Charles Stelding, 1415 South Baxter, Tyler, TX 75701.

Have you wanted to really put ARCHIVE to work (the database program that comes "free" with a QL), without a lot of programming headaches? You may want to check out the ARCHIVIST and MAILMERGE software packages from Ark Distribution, Corve Farmhouse, Chale Green, Ventnor, Isle of Wight, U.K., PO38 2LA. Write for information and pricing or contact your local QL dealer for a demonstration.

MIDWEST TS COMPUTER FEST

A Huge Success - AGAIN!

by Joe Willlamson

May 2nd and 3rd was the peak time of the year for Timex and sinclair enthusiasts who once again converged for a weekend of fun and info gathering on our beloved computers. This year Indianapolis, Indiana was the site for the 2nd Midwest Timex Sinclair Computer Fest. If you were not there, you missed the best one yet!

More than 45 dealers and user groups displayed their wares in over 6000 square feet of space. There were two separate seminar rooms with scheduled seminars running all day in both rooms. There was also a "swap shop" room were TS users exchanged and sold their unused equipment. A banquet was held on Friday night before the Fest to get all the dealers and user groups aquainted which was enjoyed by all.

More than 700 people were in attendance with family members either enjoying the show or the excellent hospitalities of the Holiday Inn or even Indianapolis itself including the Speedway which was having time trials that weekend. All in all, this show was more than twice the size of last year's show. As last year, people from all over the US were there as well as from Canada, England, Mexico, and El Salvador.

The entire line of Timex & Sinclair computers were represented as well as some clones such as the Thor (a QL clone) and the PC 8300 (a TS 1000 clone). Many semi-supporters were also there including representatives from CTN magazine, Computer Shopper, compuserve, PC Pursuit, and The White Church Cabin who sold buttons commemorating the Computer Fest.

There was a shift of emphasis this year to the QL which seems to be doing much better than alot had expected last year. Sharp's, Brice Road Pharmacy, Variety Sales, C. W. Associates, Curry Computer, Quantum Computing, Markel Electronics, RMG Enterprises, Time Designs, Sync Ware News and Quantum Levels all had excellent products available for the QL at the show. A+ Computer Response was also there showing their support for the QL.

The 2068 was not lacking in support either. Several new products were demonstrated and sold. RT Mnemonics, Variety Sales, The Widjup Co., Foote Software, Byte Power, JRC Software, WMJ Data Systems, Grey & Clifford, Curry Computers, RMG Enterprises, Novelsoft, Aerco, Zebra Systems, Knighted Computers, Semper Software, EZ Key, E. Arthur Brown, The John Oliger Co., Leake Software Development, Larkin, Time Designs, and Syncware News all had products or catalogs available. Most of the seminars covered the 2068 and it seems that this year the trend is toward telecommunicating with the 2068 instead of just playing games. Also, desktop publishing on both the 2068 and QL seems to be popular this year.

The ZX81/TS1000/1500 had its own following with such companies as WMJ Data Systems, Sirius Ware, Zebra Systems, Silicon Mountain Computers, The John Oliger Co., Semper Software, EZ KEY Thomas B. Woods, Syncware News, and Time Designs represented at the show.

The User groups were also well represented and are becoming a key to the continued support of our computers. The Indiana STUG, Greater Cleveland SUG, Chicago Area TUG, Sinclair Milwaukee UG (SMUG), Capital Area TSUG (CATS), North East Florida TSUG, Sinclair Louisville UG (SLUG), Tampa and ST. Pete Area Members TSUG (TASBAM), The San Diego UG, SAF UG, and the Quanta QL users group from England. User group representatives from all over the country were there to share and gather information.

There were many door prizes donated which made it possible to have about eight prizes given away each hour! Some of the door prizes given away were the QL, software packages, service manuals, subscriptions, gift certificates, books, and newsletters. Everyone I spoke with said that they enjoyed the show and will return again next time. Next time is already on the drawing board and will be a reality. Also in the works is a show to be held next March in the Orlando Florida area. The producers of the Midwest TS Computerfest will help the North East Florida TSUG and the TASBAM group put on the March '88 show which has already received tremendous support from the dealers present at the last show.

This is an excellent opportunity for everyone to come down and bring their family for a nice vacation in Florida. The site for the show will be very close to Disney World, Sea World, Circus World, and other major attractions. Also, this is off season time and vacation packages will be available at reasonable rates.

For more information, contact Eric Johnson, 249 N. Harden Ave, Orange City, FL 32763. A BBS will be set up at his address to gain more information soon. Actual date of show will be announced soon. Start making plans now!

The show was very well organized and proper credit should be given to all those envolved. The Show was put on by a non-profit enterprise comprised of various user groups and individuals in the Midwest. Frank Davis is the producer and TS Founder. He and the following people should be applauded for putting on an excellent show.

Paul Holmgren	Executive Chairman
Willie Jones	User grp. Coordntr/Registration Chairman
Ralph McCrum	Swap Meet Chairman/ Door Prize Coordntr.
Basil Wentworth	Seminar Chairman
Carol Davis	Computer Fest Consultant/Treasurer
Rhonda Jones	Special Assistance
Tim Woods	Program Booklet Coordinator

1987 Computer Fest Committee Members:

Jack Payne
Tom Burt
Hilda Burt
Bill Bell
Mike Felerak
Frank Duncan

Also, special thanks should be given to the Holiday Inn North of Indianapolis and many others who made the 1987 Midwest TS Computer Fest possible. I hope I covered everyone that was there!

Stewart Newfeld of Zebra Systems

WHO was there? WHAT was there?
Another look at the 1987 Sinclair Extravaganza...

Reported by Tim Woods

Most "sophisticated" computer people who can even remember the Timex Sinclair, would like to think that it dried up and withered away. But remember the old saying "you can't keep a good man down"? Certainly holds true with the Timex Sinclair community. Participation of both dealers and users at the 1987 Midwest TS Computer Fest held on May 2nd and 3rd in Indianapolis, Indiana, re-affirmed and demonstrated that there still is tremendous interest in Sir Clive's computers.

Something must be said of this phenomena of getting behind an "orphan" computer and supporting it long after the manufacturer has severed ties and even forgotten it exists. (Evidence of this is seen elsewhere; some 10,000 die-hard TI994/A fans recently held a computer fair in Chicago. One enterprising individual is manufacturing TI clones. There is even a large support group for the Adam and another one for the IBM PCjr.) It is like a silent revolution, where the consumer takes action into his or her own hands. In a conversation with one long-time TS supporter at the Fest, he told me, "I have grown to be comfortable with my Timex system and it's kind of a challenge to find where I can still get programs and hardware for it. Sure there are faster, more powerful machines, but I am very happy with what I have. Why should I go out and plunk down some bucks for an IBM clone or an Atari ST, when I haven't even explored all of the potential of my own machine? And I'm having a lot of fun too!"

Several new items were displayed or announced for the first time ever at the TS Computer Fest. Many of the exhibiting dealers had large booths with monitors set up to demonstrate their wares.

Mark Steuber from Sharp's Inc., of Mechanicsville, Virginia (who also happens to be the author of WAR IN THE EAST, a program that has sold quite well in the U.K. recently), had several new offerings for the QL, along with some news of other developments. Most important perhaps is that a new board called the TRUMP CARD, will available by the time you read this, for $299.95. And get this—along with a full-featured disk interface you also get additional RAM that will boost your QL's memory to 900K!! At the Sharp's table were some hardware up-grades from Miracle Systems and Sandy (including the SUPERQBOARD and SUPERMOUSE). A new keyboard replacement for the QL called the Schon Keyboard was also shown. Reportedly it only takes about three minutes to install, and is priced at about $90. It had the feel of a very good typewriter keyboard. Mark said that several of his customers are interested in the new CP/M operating system from Ultrasoft, and that he has it on order. This should open a vast amount of public domain software for the QL.

On another note, it was announced that Sharp's Inc. had bought out Knighted Computer's QL business. Knighted Computers (of Fulton, New York) will no longer stock QL products, but instead will be concentrating on the other computer lines like the Amstrad, but will also continue to support the Timex Sinclair 2068. Ray Payne, co-owner of Knighted mentioned that the QL business had been an "up and down" affair, yet the TS2068 has proven to be a very steady enterprise. They have a whole list of superb programs for the Timex that have been converted from the Sinclair Spectrum, including their latest, TOMAHAWK, a helicopter flight/air battle simulation.

Continued Next Page...

Jon Roketenetz (programmer) and Mark Pendrick of Markel Enterprises have just released "ElectriQL Desk".

The CST THOR, a QL clone is shown by Curry Computer.

Joe Williamson at the Foote Software booth.

TDM's Editor, Tim Woods chats with Mowgli Assor, the programmer of "SEKTOR 2068", a disk utility.

At the Sharp's Inc. booth: the Schon Keyboard and disk interface/RAM board from Miracle Systems.

MIDWEST TS COMPUTER FEST

Zebra Systems of Woodhaven, New York, again (as in Cincinnati last year) had the largest display of Timex Sinclair merchandise. Everything from books, Memotech interfaces, programs, and even a new Wico Trackball controller for the Sinclair QL. For months, rumors had been circulating in the Sinclair community that Zebra was "getting out" of the Timex business. But the truth is, while they do have on-going development for other lines (like the Tandy and Atari), they have committed to continue support for the Timex. They have a large inventory of related merchandise including a number of used ZX81s, ZX power supplies, and un-tested 2050 modem cards. Stewart Newfeld, manager of Zebra Systems even reported that an all new catalog would be printed shortly. I think that most Fest attendees were impressed with Zebra's attitude and confirmation of support for TS users, which greatly differs from some of the rumors that were floating around recently.

It was a real joy to see that Curry Computer could make it to this year's Fest (all the way from Phoenix, Arizona). They brought with them a good sampling of their Sinclair product lines, which leans heavily towards the Sinclair QL. On display was both a Sinclair Spectrum 128 (pre-Amstrad) and the CST THOR (the newest

TS Fest Committee members and helpful personages: (back row, L to R) Gary Ganger, Tom Burt, Hilda Burt, Paul Holmgren, Willie Jones, Basil Wentworth, Bill Bell, and Frank Duncan. (Front row, L to R) Frank Davis, Carol Davis, Jack Roberts, and Rhonda Jones. A map depicted areas where Fest attendees came from.

The Friday night Banquet was attended by the exhibitors and Fest committee members. It was held in the hotel's spacious ballroom, complete with crystal chandeliers. This was the perfect "kick-off" to the successful 1987 Midwest TS Computer Fest.

The food at Friday night's Banquet was superb! Salads, croissants, tenderloin tips, chicken, and world famous strawberry cheesecake were the featured fare...and the service wasn't bad either.

Editor Tim Woods and Assistant Manager Stephanie Woods of TIME DESIGNS, enjoy excellent dinner conversation with Ian Robertson (far left) from Toronto, Canada and several other TS notables.

Tom Bent answers a customers question. Tom is the editor of QUANTUM LEVELS and is the U.S. librarian for QUANTA, the London-based QL users group.

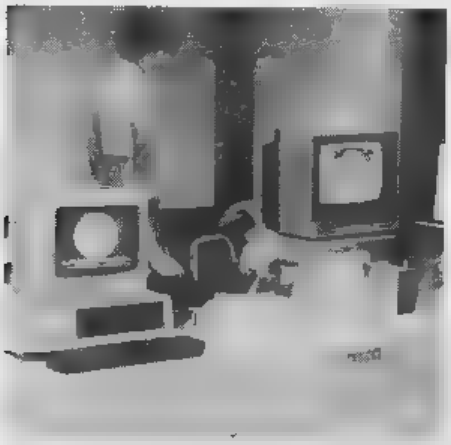
Representatives of CATS, the Washington, D.C. area TS users group, (L to R) Hank Dickson, Audrey and Bob Curnutt, and Ruth Fegley.

QL clone from the U.K.). They also had a video showing highlights from their recent trip to a Personal Computer Show in England. Rob Curry demonstrated how the Psion Organizer works. It's a hand-held pocket computer, that can download and upload QL data via a serial cable. This is one powerful little device to watch, as versions are available for IBM compatibles and is even reported to be adaptable to Lotus 1-2-3.

MIDWEST TS COMPUTER FEST

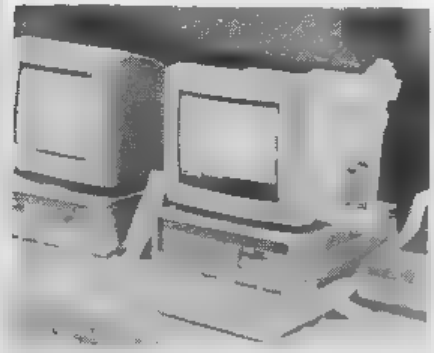
Mark Fendrick was back again this year representing Markel Enterprises, and had teamed up with Peech II Productions (Jon Rokentenetz) to produce a new QL program called ELECTRIQL DESK. It has an icon-driven menu and is similar to "sidekick" programs that contain several useful functions in one software package. ELECTRIQL DESK is priced at $24.95.

Continued On Page 10



Chris Raynak mans the Greater Cleveland Sinclair Users Group booth. The group had an impressive display of graphics.

Mark Steuber of Sharp's Inc., explains how the latest QL hardware and software innovations operate.

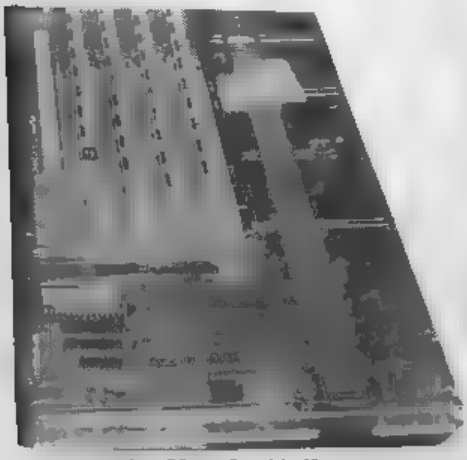


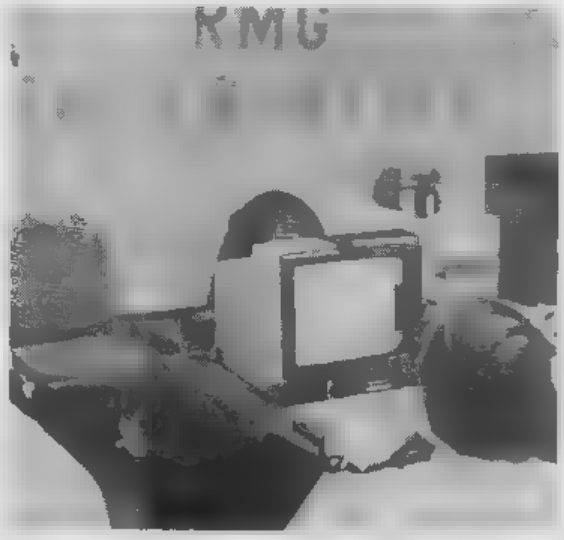
It's the Amstrad PC1512, an IBM clone, displayed here by a local Amstrad rep. Complete systems start at $800.

Ariel Frailich, programmer for Novelsoft of Toronto, Canada, dons his "expert" cap to promote their new program ZXPERT.

Staff of Quantum Computing on hand to answer questions: Monica Merel (General Manager) and Colin I. Cooke of London, England (Director of International Operations).

Smiling Debby Curry, co-owner of Curry Computer, reported that Spectrum software sold very well at the Fest.





Left: "One of a kind"--a transparent TS2068 with a FootePrint Interface in the cartridge dock. Center: Dick Cultice, a member of SMUG from Wisconsin, at the helm of his TS2068 system including a full-blown Aerco FD-68 floppy set up. Right: At the RMG Enterprises booth, Rod Gowen demonstrates the new Larken TS2068 disk system.

MIDWEST TS COMPUTER FEST

There were many other exhibits geared to the QL including: A+ Computer Response, C.W. Associates (who had a striking all black dual quad-density disk drive system for $239), Don Banard of Sinclair Network (has a product guide on microdrive cartridge), Variety Sales (demoed networking and a QL-based BBS program), QUANTUM LEVELS magazine, and Quantum Computing.

Frank Toemay of Quantum Computing is a rather interesting fellow that has launched an ad campaign for the Sinclair QL in many elite PC engineering publications, to go neck-to-neck with PC's and other 32 bit processors. Frank also offers two software packages of "freeware" (Freeware I and Freeware II) at no charge, by just sending a blank microdrive cartridge for each, or sending $2 for each package to cover the cost of a cartridge and postage. Each freeware contains utilities and other programs for the QL.

Two new QL books were on display: "Taking The Quantum Leap: The Last Word On The Sinclair QL" by Mike de Sosa was featured at the Time Designs Magazine booth, and "Archive Master" by Executive Workshop was available for viewing at RMG Enterprises booth.

The most interesting 2068 display at the Fest was clearly (pun intended) a completely transparent, plastic moulded TS2068, which was obtained from the Research and Development Department of the Timex Computer Corporation shortly after that division shut down. Inside this rare bird, one could easily see the all-socketed printed circuit board, with EPROMS replacing the usual factory ROMs. A user from Florida is the lucky owner of this collector's item. He also obtained Timex-designed stick-on keyboards for the TS1000. These would turn the normal membrane keyboard into small calculator type keys. These have the Timex logo on them, and would have been sold through Timex dealers.

Jerry and Til Champkif of AERCO from Austin, Texas, were back again this year supporting the Timex Sinclair 2068 with their FD-68 disk drive and RP/M systems. New for this year, was the long-awaited Spectrum "boot" disk which allows the FD-68 user to run Spectrum software. On Saturday evening, an exclusive Aerco users meeting was held. Jerry Champkif announced that Aerco was dedicated to finishing the FD-68 DOS, and that completion of this goal was not too far off.

Novelsoft from Toronto, Canada, had an impressive display of three monitors that gave a continuous demo of their popular software packages for the TS2068 and the Spectrum, including TIMACHINE, ARTWORX 1.1, The WORX!, and a new release called ZXPERT. It is an expert system and also (with the accompanying booklet) a study in artificial intelligence-type environments. On hand to answer questions were Novelsoft programmers David Ridge and Ariel Frailich.

Ed Grey of Grey & Clifford Computer Products was at his terminal demonstrating a Z-SI/O RS232 serial port card and the SPECTERM-64 software package, which will allow the TS2068 user to operate any 1200 baud modem. Grey & Clifford is also a dealer for PC Pursuit, and applications were available for signing up. A special separate drawing was held at the booth, and these lucky individuals won G & C merchandise: Don Walterman (MI) and John Kemeny (MA) both won Z-SI/O cards, and John Coffey (IN) and Oscar Sensabaugh (TX) were winners of SPECTERM-64 software.

Ed and Heddy Grey of G & C Computer Products

Joe Williamson of Foote Software from Gainesville, Florida has one of the best printer interface values around for the TS2068. The FootePrint is a quality board that fits in the cartridge dock, and is compatible with print driver software for the Aerco/Oliger and both Tasman B and C. The $45 price includes the card, ribbon cable, and software on cassette. There is even a spare socket and switch, to run either a Spectrum ROM, or any EPROM based software (like Zebra's OS-64).

Other Timex Sinclair 2068 displays included: Rod Gowen of RMG Enterprises (demonstrating the new Larken Disk System and many original software packages that RMG exclusively markets), Vern Tidwell of RT Mneumonics (the co-author of SPRITES 2068 demoed some new programs and displayed special AROS versions of each), the WDJUP Co. (offered a new catalog featuring a number of software packages), Stan Lemke of Lemke Software Development and regular contributor to TDM (demonstrated his new desktop publishing package for the TS2068 called PIXEL PRINT), Pete Fischer (had a new and expanded version of his GUIDE TO TS TELECOMMUNICATIONS), John Coffey of JRC Software (showed DIAMOND MIKE and other programs), and both Jeff Moore (editor) and Tom Woods (publisher) were on hand from SYNCWARE NEWS.

Speaking of Tom Woods, he has been shifting his talents a bit lately and has been involved in developing a new program for use on PC compatible machines called FINDEX. It is a database inspired by his Pro/File series but makes use of disk drives, 80 col. video, and greater memory capacity. One interesting feature of FINDEX is that it can transfer data files from other computers (such as the Timex) into its database through an RS232 serial interface. The program will be marketed by the E. Arthur Brown Co. as "shareware".

While not as plentiful, there were some displays of interest to ZX81/TS1000/TS1500 users, which included: Mike Amling of Semper Software (featuring the Partial Pascal software package), Dave Woods of SiriusWare (and author of a popular machine code book for the ZX81), Ted Sobel of The White Church Cabin (brought along an interesting industrial applications display based on a TS1500, some custom software and extensive control hardware). Many dealers had software packages for the TS/ZX based machines, and SYNCWARE NEWS and TIME DESIGNS had magazines and other publications available.

There were many other individuals and groups who participated and worthy of noting: John Oliger of The John Oliger Company, Dave Rothman (SYSOP for CompuServe), Mike Davis and Duane Malburg of MDM, Kurt Casby (programmer of LOADER V and CASBOARD 2068), Dick Kelly of Kelly's Office Products (an Amstrad dealer), Dick Thatcher of Howard W. Sams Publishing, Gary Soloman of Brice Road Pharmacy, John Kemeny of the Boston Computer Society, Gary Ganger (President of DMA), Dave Franson (programmer of Extended Paint and editor of T.O.P.S.), Pat Spera (SYSOP for CompuServe), Jack Roberts (1986 TS Computer Fest Chairman), and the many members and representatives of the more than 12 TS Users Groups present at the Fest. I am sure that some person or group was accidently omitted from this list...for this I sincerely apologize. Perhaps another article could be written on things that were left out of this one.

Based on the success of the Second Annual Midwest TS Computer Fest, there will be another one next year. Several ideas were suggested, such as three separate shows (one in the West, one Midwest, and one on the Eastcoast). Even more localized computer shows on a smaller basis have been discussed for Florida and possibly the Northwest. I know that I will attend at least one if not all of the proposed TS Computer Fests. If you hesitated about Cincinnati, and procrastinated this year about Indianapolis, perhaps next year will be your chance. See you there!

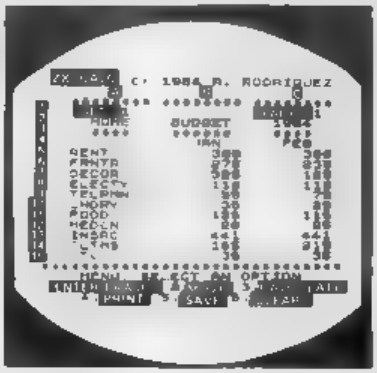
Note: Photography at the 1987 Midwest TS Computer Fest by Joe Williamson and Tim Woods. Please read Joe's accompanying article on the Fest.

POWERFUL AND INEXPENSIVE BUSINESS SOFTWARE FOR ZX81, T/S1000 and T/S1500 COMPUTERS

ZX-CALC

A word processor is to a computer user what a typewriter is to a typist, except that the former has more advantages than the latter. ZX-Text can operate in 16-64K RAM providing from 1300 to 6500 words per document. It features 5 different options: write, read, edit, print, save and clear text. Text is written on a per-line basis with quick speed and with horizontal back-space and delete capabilities being available. You can also access the editor directly from write mode and vice-versa. Text can be proof-read on a per-line basis allowing for enough time to determine if any editing is needed. The text editor allows a line of text to be deleted, inserted, replaced and listed for editing. You may also change a word or expression within a line, stop or start text while it is scrolling up the screen, begin reading text from the first line of the file, re-enter write mode from the editor, return to the main-menu, or create a window so that you can read-edit two files simultaneously. The print option takes text displayed in 30-column format on the screen and outputs to either the ZX/TS printer. (With Memotech's Centronics Parallel interface 80-column and lower/higher - case output is possible.) Files may be saved on tape cassette with the use of one single command, or by the same token they can be erased from memory / storage so that the full capacity of the program can be used for other purposes such as composing letters, reports, articles, memos, standard forms, instructions, ads, graphs, telephone directory, lists of customers, members, friends, etc. Also copies of files are always less expensive and easier to run than using a photocopier. Other advantages are savings in time, paper, ink, correcting mistakes and adding afterthoughts more efficiently than doing them through either handwriting or using a typewriter.

$16.95



An electronic spreadsheet calculator is the fundamental basic tool for summarizing, reporting and analyzing in matrix form any accounting, mathematical, or scientific manipulation of numbers. ZX-Calc operates in 32-64K RAM and affords a maximum of 3360 characters / spreadsheet. The entire matrix consists of 15 columns (letters A-O) and 30 rows (numbers 1-30) with 8 characters / cell. Unlike other popular ESCs, ZX-Calc uses in calculations and within cells all 14 math functions on the ZX-81 / TS1000. It offers a unique "SUM function that totals one or more rows / columns simultaneously. Parenthesis can be used within equations. There is no fixed limit on how many equations may be entered. Formulas may be stored in all 420 cells of the spreadsheet. The display affords 15 rows / colums. Loading of data into more than one cell can occur across / down one or more row / column simultaneously. With vertical windowing you can arrange a set of columns in any order or practice using fixed- variable-alignment display formats. The menu offers 6 options: enter / erase, move, calculate, print, save and clear the spreadsheet. Enter / erase allows the entering, deletion or data alignment within a cell through the use of a mobile cursor. With the move option you may move around the entire spreadsheet to access any row, column or cell. The calculate option allows you to enter labels, values or formulas into a cell or write and enter equations that will act upon the data already within the spreadsheet. You can also enter bar graphs into a cell in this option. Absolute / relative replication, down / across a column / row, is also allowed by this option. Also this option allows the automatic calculation of the entire spreadsheet with one single command. Print allows you to output to either the ZX / TS printer the entire spreadsheet by column-sets and row-pages through use of the COPY command. The entire spreadsheet may be saved on cassette tape or you may clear all data from it or erase the program from RAM entirely. The most salient advantage provided by an ESC over specifically vertical applications software is that an ESC provides a reusable framework with which you can compose any specific financial model rather than just be limited to only one statically fixed format for storing, displaying and manipulating numerical data.

$16.95

Time management is an important aspect of any serious business and personal agenda. Planning how to spend our time leaves us better prepared before and while we are spending it and we remain better organized after we finish spending it. ZX-Calendar operates in 16-64K RAM affording 25 appointments in 16K, 100 in 32K or 180 in 48K and 64K. Each appointment record holds a maximum of 220 characters. The main menu includes enter, search / check / sort, change, save, clear and print any and all appointments made on a specific date or with any party. Output to either the ZX / TS printer is permissible. This program will permit you to remember to do something or to be somewhere important by cataloging your answers to six questions that you must account for in order not to waste time when it is scarce: when, with whom, at what time, for how long, where and what are you going to discuss and conclude when you get together with someone else? The program lets you permanently originate, record, classify, search, sort, calculate, modify, summarize, obtain a written report and store your answers to the preceding questions so that you will not forget what you decide to do with your time. This program identifies your time according to when you are going to spend it and with whom you are going to share it. Through these forms of labeling appointments you are able to verify or modify how your time is budgeted without wasting ink, paper or more time trying to remember what you said to yourself or what someone else said to you or where you placed certain written messages that you now can't find. With this program you will know where you can find exactly what you need to know about where you want to and have to be or where you have been before you get and after you got there. Thus, ZX-Calendar will let you plan your time so that you will never have to worry about what is ahead or what came before, for you will always know, by using it, to never be caught astray by any time-frame.

$16.95

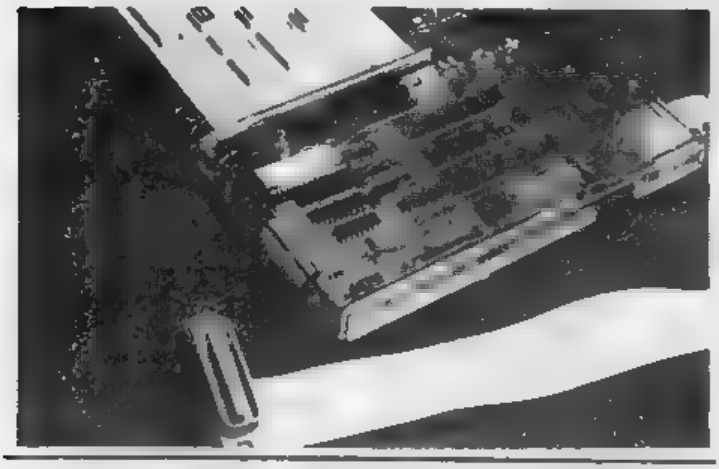
$3.00 SHIPPING AND HANDLING / PROGRAM

A.F.R. SOFTWARE - 1605 Pennsylvania Avenue, No. 204 - Miami Beach, Florida 33139

DEALER INQUIRIES WELCOME **(305) 531-6464** FLORIDIANS ADD SALES TAX

L-1 © 1984

FOOTE SOFTWARE



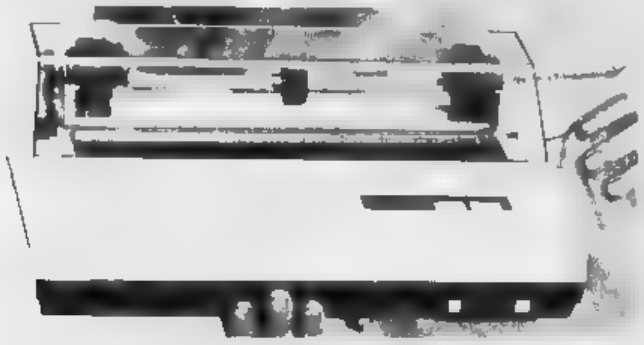
The FOOTE PRINT PRINTER INTERFACE

- for Centronics parallel printers
- works in both 2068 and Spectrum mode
- compatible with OS-64 & Spectrum emulators
- EPROM socket and on/off switch on board
- works with **both** Tasman and Aerco driver software
- plugs into cartridge dock—door completely closes with cable running back under computer
- frees up rear edge connector allowing other peripherals to be used; less chance of a crash
- print driver software for LPRINT, LLIST, and COPY included for 2068 and Spectrum modes

FootePrint Interface w/software & cable $45.00
FootePrint with OS-64 option included ..$65.00
Bare board & instructions only $20.00
Cable only for use with bare board $15.00
Zero Insertion Force Socket option add $10

SOFTWARE	TS2068	TS1000
Badgammon (Backgammon)	$12.95	
Advanced Math (Calculus)	$12.95	$7.95
Calorie Counter	$9.95	$5.95
U.S.A. (Pres. & States & Capa.)	$9.95	$5.95
Gambler (poker)	$9.95	
CHR$ (char. & graphics generator)	$12.95	
Hangman & TIC–TAC–TOE	$5.95	

Brother M1109 Dot Matrix Printer, compact, low noise, 100 CPS, both Parallel and Serial interfaces, multiple typestyles with near letter quality print mode and 4k memory buffer, comes with tractor feed unit..... **$249.95**
QL or Zebra FDD cable for above: **$17.00**



THE BEST OF

S U M

of SINCLAIR/TIMEX USERS MAGAZINE

A compilation of the best articles, reviews, programs, and hardware projects

The Best of SUM

Some sample articles include: Building Your Own Spectrum Emulator, Repairing Your TS-1000, Word Processing Reviews for the 2068, UDGs on the TS-1000, Extensive Review of the Zebra Disk System, Adding a Keyboard to the 2068, and Enhancing the A & J Microdrive. 112 pages

Price: $11.95

The Best of SUM, Part II

Art cles include Building an EPROM Programmer, Sprites on the 2068, Adding RGB to 2068, QL Word Processing, What s Available for TS-1000 and much more. 60 pages

Price: $7.95

FOOTE SOFTWARE P O Box 14655 - Gainesville, FL 32604 904, 462-1086 (6 pm - 9 pm EDT)
All prices are pre-paid and include shipping charges. Florida residents must add 5% state sales tax

CK Type

A NEW "STANDARD" FOR "FOOL-PROOF" TYPING OF TS2068 PROGRAM LISTINGS IN MAGAZINES

BY STAN LEMKE

One of the really great things about attending the 1987 TS Computer Fest was meeting and talking to all the great Timex/Sinclair enthusiasts! What's more, all the people that type-in and use programs presented here in TDM. Several people commented on the difficulty of typing in these programs, and then debugging them. The longer the program, the more difficult the task. As an aid in this endeavor, I've created CK TYPE.

CK TYPE will give a short analysis of every line in your BASIC program. It provides the following information: LINE NUMBER, number of BYTES used to create the line, and a CHECKSUM of these bytes. If a program is submitted and the CK TYPE output is provided for the listing, you will be able to type in your copy, run CK TYPE on it, and very quickly know which line(s) might need corrections. (Note: OUTPUT is routed to the screen and to the TS2040 printer with LPRINT, simultaneously.)

The LINE NUMBER lets you check line-by-line the accuracy of your listing. A line is the BASIC program line following a number (1 to 9999), and may be 1 word, or a full screen in length.

Number of BYTES used is dependant on the number of characters you type (including blanks and embedded attributes) and helps you find those missing "spaces".

CHECKSUM helps you find typing errors such as misspelled variables, errors in data statements, and even transposed characters. It is not "fool-proof", as compound errors could offset each other, but it sure helps. (Note: CHECKSUM for upper and lower case letters is different, so be sure to use the same case as the listing you are typing!)

Listing "A" gives you the CK TYPE Basic program. Simply type it in as is. If you do not have TIMACHINE (the compiler from NOVELSOFT), then you can delete/skip lines 9987 to 9990 as these are TIMACHINE directives. When you are sure the program is typed correctly, save it to tape with: SAVE "CK TYPE" [ENTER].

Listing A

```
9987 REM ' LIST
9988 REM ' LPRINT
9989 REM 'INT +START STOP,BYT LI
NE,LENGTH,SUM,ODD,I
9990 REM ' OPEN #
9991 LET start=256*PEEK 23636+PE
EK 23635  LET stop=256*PEEK 2362
8+PEEK 23627  LET byt=start
9992 LET line=0  LET length=0
ET sum=0
9993 LET line=256*PEEK byt+PEEK
(byt+1)  LET byt=byt+2
9994 LET length=256*PEEK (byt+1)
+PEEK byt  LET byt=byt+2
9995 LET odd=1  FOR i=1 TO lengt
h  LET sum=sum+(odd+1)*PEEK byt
 LET odd=NOT odd  LET byt=byt+1
 NEXT i
9996 PRINT line,TAB 6,"  ",lengt
h,TAB 13,"  ",sum
9997 LPRINT line,TAB 6;"  ";leng
th,TAB 13,"  ",sum
9998 IF byt<stop THEN GO TO 9992
9999 STOP
```

Listing B

```
9987     4       994
9988     4       962
9989     41      4728
9990     4       936
9991     96      8966
9992     43      4165
9993     53      5151
9994     55      5569
9995     89      10304
9996     46      4502
9997     46      4460
9998     22      2843
9999     2       465
```

Listing "D" is a POKE table to create a machine code version of CK TYPE (if you do not have TIMACHINE). The first column defines the memory location of the number in column 2, column 1 + 1 is the memory location of column 3, column 1 + 2 is the memory location of column 4, etc. You can create your machine code version by poking the appropriate memory locations with the corresponding values: POKE 64888,205 [ENTER]. POKE 64889,58 [ENTER]. POKE 64890,255 [ENTER]....SAVE this to tape with: SAVE "CK TYPE" CODE 64888,460 [ENTER]. To use, LOAD the BASIC program you wish to check, LOAD "CK TYPE" CODE 64999,460 and run with RANDOMIZE USR 64888 [ENTER].

I hope that CK TYPE is widely used in the TS2068 community as a way of easing the pain associated with typing in program listings! (Editor's Note: Please pass the word around about CK-TYPE, especially those of you connected with TS user groups and newsletters. Stan has come up with a standard that is common in many other PC magazines. Additional copies of this magazine can be purchased by those interested in CK TYPE for just $3 postage paid each.)

The program is then ready to RUN. Type RUN [ENTER]. The output it creates will match listing "B" if you have no errors. The first column from listing "B" is the LINE NUMBER, column 2 is the number of BYTES used to create the line, column 3 is the line CHECKSUM value. Non-TIMACHINE versions would start with the line number 9991. To use this version with other BASIC programs, make sure that your other program does not use lines 9991 to 9999. Then MERGE CK TYPE with your other BASIC program, and RUN 9991 [ENTER].

If you use the TIMACHINE compiler to compile this program, it will produce listing "C" and a machine code routine that will be located at 64888 and be 460 bytes long. SAVE this to tape with SAVE "CK TYPE" CODE 64888, 640 [ENTER]. To use this, LOAD in your BASIC program, LOAD in the machine code program with: LOAD "CK TYPE" CODE 64888,460 [ENTER], and run with RANDOMIZE USR 64888 [ENTER].

Listing C

```
LINE 9990  +0
LINE 9990  64888 #FD78

RT20      65249 #FEE1
RT31      65257 #FEE9
RT47      65263 #FEEF
RT91      65275 #FEFB
RT104     65283 #FF03
RT111     65323 #FF2B
RT123     65331 #FF33
RT124     65338 #FF3A
RT125     65343 #FF3F
start..... POSINT    65348 #FF44
stop...... POSINT    65350 #FF46
byt....... POSINT    65352 #FF48
line...... POSINT    65354 #FF4A
length.... POSINT    65356 #FF4C
sum....... POSINT    65358 #FF4E
odd....... POSINT    65360 #FF50
i......... POSINT    65362 #FF52

TIME MACHINE ©1986 Cameron Hayne

M/C  460 BYTES
+ 20 BYTES FOR M/C VARIABLES
(BASIC WAS 557 BYTES)

SAVE "m/c"CODE 64888 460
LOAD "m/c"CODE 64888
```

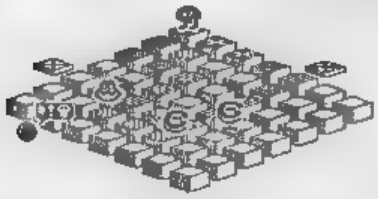

Program Continued Next Page...

Listing D

```
64883  205  58  255  33  66
64888  255  34  108  92  33
64893  0  1  229  33  84
64898  92  110  38  0  209
64903  205  3  255  229  33
64908  33  92  110  38  0
64913  209  25  34  68  255
64918  33  3  1  229  42
64923  76  92  110  38  0
64928  209  205  3  255  229
64933  33  75  92  110  38
64938  0  209  25  34  70
64943  255  42  68  255  34
64948  72  255  33  0  0
64953  34  74  255  33  0
64958  0  34  76  255  33
64963  0  0  34  78  255
64968  33  0  1  229  42
64973  72  255  110  38  0
64978  209  205  3  255  229
64983  42  72  255  35  110
64988  38  0  209  25  34
64993  74  255  42  72  255
64998  35  35  34  72  255
65003  33  0  1  229  42
65008  72  255  35  110  38
65013  0  209  205  3  255
65018  229  42  72  255  110
65023  38  0  209  25  34
65028  76  255  42  72  255
65033  35  35  34  72  255
```

```
65043  33  1  0  34  80
65048  255  33  1  0  34
65053  82  255  229  42  78
65058  255  34  84  255  225
65063  195  89  254  42  80
65068  255  229  42  80  255
65073  35  229  42  78  255
65078  110  38  0  209  205
65083  3  255  209  25  34
65088  78  255  42  80  255
65093  205  251  254  34  80
65098  255  42  72  255  33
65103  34  72  255  42  82
65108  255  35  34  82  255
65113  237  91  84  255  25
65118  167  237  82  210  48
65123  254  42  74  255  205
65128  233  254  33  6  0
65133  205  51  255  205  239
65138  254  2  0  58  32
65143  42  76  255  205  233
65148  254  33  13  0  205
65153  51  255  205  239  254
65158  2  0  58  32  42
65163  76  255  205  233  254
65168  62  13  215  205  63
65173  255  42  74  255  205
65178  233  254  33  6  0
65183  205  51  255  205  239
65188  254  2  0  58  32
65193  42  76  255  205  233
```

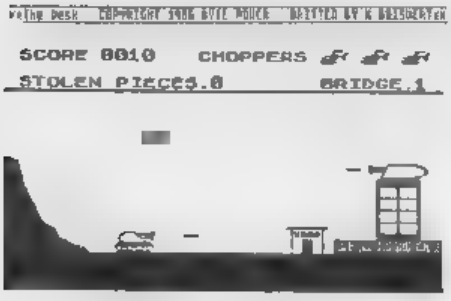
```
65198  254  33  13  0  205
65203  51  255  205  239  254
65208  2  0  58  32  42
65213  73  255  205  233  254
65218  6e  13  215  205  50
65223  255  42  7e  255  2..
65228  91  70  255  205  e5
65...  254  1e4  131  194  ...
65238  253  33  22  43  ...
65243  201  33  2e  43  217
65248  201  175  237  82  103
65253  111  208  44  201  205
65258  4..  255  195  161  49
65263  225  78  35  70  35
65268  84  93  9  229  195
65273  21e  33  125  180  33
65278  0  0  192  44  201
65283  6e  32  187  58  ..
65288  122  167  32  9  67
65293  e35  108  184  200  25
65298  16  253  201  77  68
65303  33  0  0  62  16
65308  203  35  203  18  48
65313  1  8  41  61  32
65318  245  178  240  9  201
65323  175  95  85  76  71
65328  195  116  46  62  23
65333  215  125  215  215  201
65338  62  2  195  49  18
65343  62  3  195  48  18
```



Q*ber

Q*ber in action!

PRESIDENT APPOINTMENTS FOR EACH MONTH



CONSTRUX IN ACTION!

ST CLASS MAGAZINE

BYTE POWER

A major breakthrough for the T/S 2068, BYTE POWER, a highly sophisticated computerized magazine on cassette.

No longer will you have to type in long, fastidious programs...JUST LOAD AND RUN!

BYTE POWER is the ultimate magazine for the T/S 2068. Each tape is full of programs, a real T/S 2068 owner's dream.

BYTE POWER is a software based magazine with over 130 programs per year, most of them in fast Machine Language. Programs such as TALKING CLOCK, EASYword word-processor, GRAND PRIX a 3D race, and SSP (Small Screen Painter) are some examples of the high quality programs published in BYTE POWER Magazine.

BYTE POWER brings you this quality programming at a very small cost, LESS THAN 39 CENTS A PROGRAM, based on 1 year subscription.

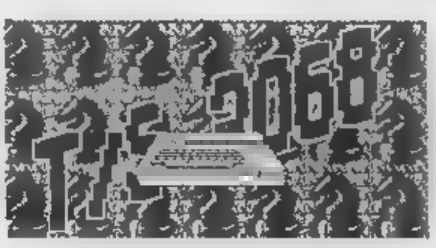
Plus you get Software and Hardware reviews and tips to help you design better programs!

1 ISSUE (SAMPLE, BACK ISSUE).....$5.50
6 MONTH SUBSCRIPTION (6 ISSUES).....$29.99
1 YEAR SUBSCRIPTION (12 ISSUES).....$49.99

U.S. FUNDS ONLY

SEND CHECK OR MONEY ORDER TO

BYTE POWER
1748 MEADOWVIEW AVENUE,
PICKERING, ONTARIO, CANADA L1V 3G8



☐ 1 ISSUE ☐ 6 MONTHS ☐ 1 YEAR

NAME:_____

ADDRESS:_____

CITY:_____

STATE:_____ ZIP:_____

Sketchit - G

by Warren Fricke

SKETCHIT-G is an artist type of program that exploits the many capabilities of the TS2068 computer to produce graphics of all sorts on the visual screen. These displays can be copied by the TS2040 printer and/or saved on tape for future recall and merging. The Timex computer has several resident functions that can be utilized for this purpose and SKETCHIT-G is designed to coordinate all of these capabilities into a single, unified program.

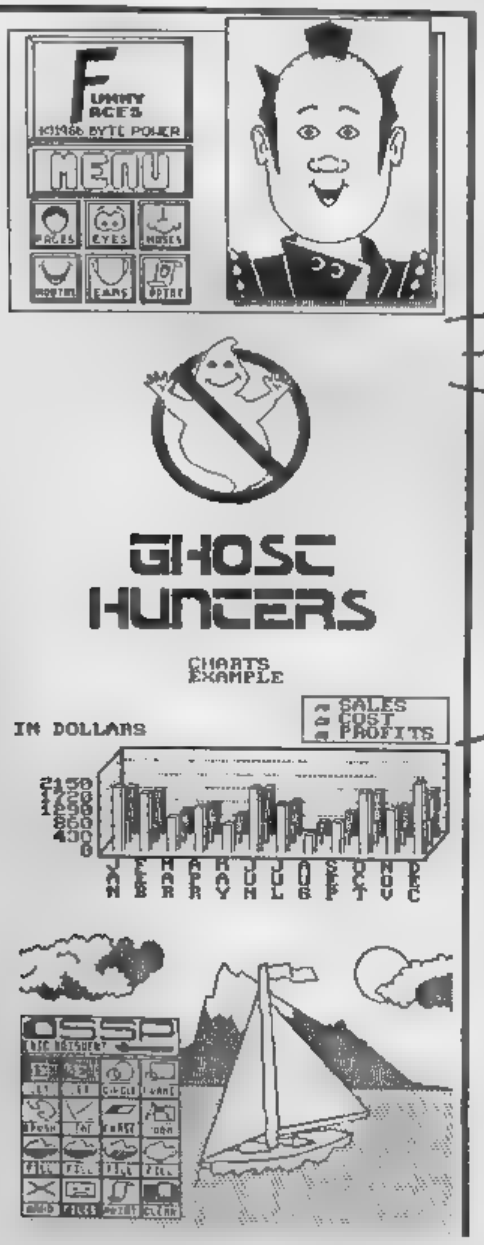
The principal routine of the program is one of drawing straight lines by plotting pixels in any of the EIGHT cardinal directions, simply by touching one of a selected cluster of eight letter keys. Most artist type of programs only go this far. Although useful in an over-all drawing program, this feature is limited in what it can do. We need more.

Straight lines in directions other than the cardinal ones can be produced by using mathematical considerations, and SKETCHIT-G does this to produce lines in two other ways: a) by using the pixel coordinates of the line terminals or, b) by using the coordinates of one terminal and its angular direction and pixel length. All three of these methods may be necessary in unhampered, graphical construction.

The TS2068 has a built-in facility for drawing circles and arcs of circles, but except for full and half-circles, the facility is impractical and difficult to control. SKETCHIT-G uses only that part which it does well, and relies upon mathematical concepts to draw arcs. But don't be scared off. The mathematics will be done by the computer. And, it has INPUT prompts that clearly tell you what data it needs from you and in what order.

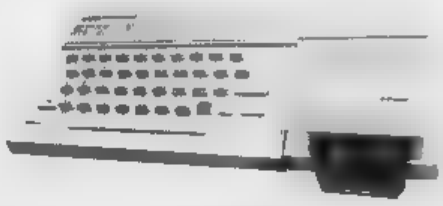
SKETCHIT-G is a user-friendly program. It includes an error trapping device that conveniently refuses to allow the program to get hung up if you should make, or try an "illegal" entry. But this can back-fire if you want to get back to the listing as BREAK does not work in the ON ERR mode. BREAK is treated like another error, and the program continues. Hence, the zero key has been set aside for an escape device. Remember this!

Continued Next Page...



Sketchit - G

```
  1 REM **   "A-44a", 6-8-86
  2 REM ** SKETCHIT-G by
           Warren Fricke
  5 BORDER 4  PAPER 6  CLS
  6 LET x=125  LET y=90  LET r=
0  DIM q$(32
 10 DIM w$(32)  DIM l$(32
 20 INPUT "Hit ENTER to continu
e ";z$
 22 ON ERR GO TO 85
 25 LET aa=PEEK 235
 30 IF aa=48 THEN ON ERR RESET
  GO TO 1000
 35 IF aa=57 THEN LET k=NOT k
 40 IF aa=83 THEN GO TO 10
 45 LET x=x+((aa=69 OR aa=67 OR
 aa=68) AND x<255)-((aa=81 OR aa
=65 OR aa=90) AND x>0)
 50 LET y=y+(aa=81 OR aa=87 OR
 aa=69) AND y<167)-((aa=90 OR aa
=83 OR aa=67) AND y>8)
 55 PRINT AT 0,6,"X = " PEEK 23
677,"     Y = ",PEEK 23678 "  "
 60 IF k=1 THEN GO TO 80
 65 PLOT INVERSE 1,x y  PAUSE 3
  PLOT x y
 70 IF aa>48 AND aa<57 THEN GO
TO 100+(aa-47)
 75 PAUSE 10  GO TO 25
 80 PLOT x,y  PAUSE 3  PLOT INV
ERSE 1,x,y  PAUSE 12  GO TO 25
 85 ON ERR RESET  BEEP 1, 25
 90 PRINT #1; FLASH 1," ILLEGAL
 MOVE "
100 PAUSE 100  GO TO 20

200 REM ** NEW PLOT COORDINATES
210 INPUT "NEW X-Value, 0 to 25
5 ";x
215 IF x<0 OR x>255 THEN GO TO
210
220 INPUT "NEW Y-Value, 0 to 16
7 ";y
225 IF y<0 OR y>167 THEN GO TO
220
230 GO TO 25

300 REM ** FULL CIRCLES
305 PRINT AT 0,0,q$
310 INPUT "X Center point = ";x
1
320 INPUT "Y Center point = ";y
1
330 INPUT "Radius = " r
340 INPUT "0 to DRAW, 1 to ERAS
E = ";b
350 CIRCLE INVERSE b,x1,y1,r
360 GO TO 25
```



```
400 REM ** LINES & SEMI-CIRCLES
405 PRINT AT 0,0,q$
410 INPUT "Starting X-coordinat
e = ";x1
420 INPUT "Starting Y-coordinat
e = ";y1
430 PLOT INVERSE 1,x1,y1
440 INPUT "X Coordinate differe
nce = ";x2
450 INPUT "Y Coordinate differe
nce =  y2
460 INPUT "0 for a line  1 for
a  ha,f-circle = ";t
470 INPUT "0 to DRAW, 1 to ERAS
E = ";b
480 DRAW INVERSE b,x2 y2 t*PI
490 GO TO 25

500 REM ** STRING$ INPUT
510 INPUT "ROW number = ";r
515 IF r<1 OR r>20 THEN GO TO 5
10
520 INPUT "COLUMN number = ";c
530 INPUT "STRING$ = ";m$
540 PRINT AT r c m$
550 GO TO 25

600 REM ** VECTOR INPUT
605 PRINT AT 0,0,q$
610 INPUT "Origin  X-coordinate
= " x1
620 INPUT "Origin  Y-coordinate
= ";y1
630 INPUT "Vector length = ";L
640 INPUT "Angle, 0 to 360 deg.
= ";A
650 INPUT "0 to DRAW, 1 to ERAS
E = ";b
660 FOR r=0 TO L
670 PLOT INVERSE b,x1+r*COS (A*
PI/180) y1+r*SIN (A*PI/180)
680 NEXT r  GO TO 25

700 REM ** ARCS of CIRCLES
705 PRINT AT 0 0;q$
710 INPUT "X Center point =  x
1
720 INPUT "Y Center point = ";y
1
730 INPUT "Radius = ";r
```

```
 740 INPUT "Initial angle (deg)
= ",a1
 750 INPUT "Terminal angle (deg)
= ",a2
 760 INPUT "0 to DRAW; 1 to ERAS
E = ";b
 770 FOR A=a1*PI/180 TO a2*PI/18
0 STEP 1.5/r
 780 PLOT INVERSE b,X1+r*COS A,Y
1+r*SIN A
 790 NEXT A  GO TO 25

 800 REM ** TITLES-COPY DISPLAY
 810 ON ERR RESET
 813 PRINT AT 0,5, FLASH 1," ENT
ER A TITLE HERE
 816 INPUT "#1 TITLE. 32 Charact
ers, max.",t$
 820 PRINT AT 0,0,t$
 830 INPUT "#2 TITLE. 32 charact
ers, max.',w$
 840 PRINT AT 21,0,w$
 850 INPUT "Get PRINTER ready &
hit ENTER",z$   COPY
 870 INPUT "INP. Y for 2nd COPY,
 else N ",z$
 880 IF z$="y" THEN COPY
 885 PRINT AT 0,0,q$  PRINT AT 2
1,0,q$
 890 GO TO 22

 900 REM ** SAVE DISPLAY
 910 ON ERR RESET
 920 INPUT "Get RECORDER ready,
hit ENTER",z$
 930 SAVE "SKETCHIT"SCREEN$
 940 GO TO 22

1000 REM ** EXIT ON ERR MODE
                and / or
              START OVER
1005 INPUT "INPUT  C=CONT. N=RES
TART  ",z$
1010 IF z$= c' THEN GO TO 22
1020 IF z$="n" THEN CLS   RUN
1030 IF z$<>"c" OR z$<> "n"
       THEN GO TO 1005
```

How to use SKETCHIT-G might best be explained by describing what the dedicated keys do. First, there are EIGHT direction keys. These are shown in Figure 1. The arrows show the direction of travel each particular key controls. The middle, or "S" key, has been selected to turn off pixel "flashing". Flashing has a long ON cycle when in the draw mode and a long OFF cycle when in the erase mode. Pixels are small. One must look close to discern this difference. Also, erase, the pixel must retrace exactly the path that it took to draw. And especially in the case with curved lines. This means that it must traverse the line in the same direction. This is because the finite size of the pixel allows only an approximation of the true line, and the same line might be drawn with other pixels in reverse.

The area set aside on the screen for these graphics is all of that covered by screen lines number 1 to 20 inclusive. Line 0 and Line 21 have been set aside for titles, remarks, etc. You may draw in these lines, but avoid doing so. It is suggested that a screen grid showing pixel coordinates plus line and column numbers be used and that the desired design be initially sketched out in pencil with terminals and angles of lines be identified by coordinates and/or degrees. This information can then be transferred to the computer line by line in almost any order. Of course one is helped a bit by having some drafting skill to do graphics, but all skill in this application is a low level requirement.

The following is a tabulation of what action each of the number keys produces:

1 - Relocates the pixel PLOT coordinates.

2 - Full circles by the CIRCLE command.

3 - Lines and half-circles by the DRAW command.

4 - STRING$. Can accept any character or group of characters, spaces, keyboard symbols, pre-fabricated UDG's, etc.

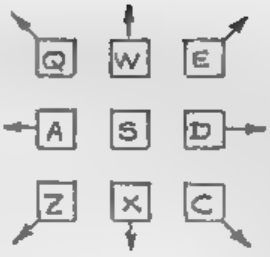


FIGURE - 1

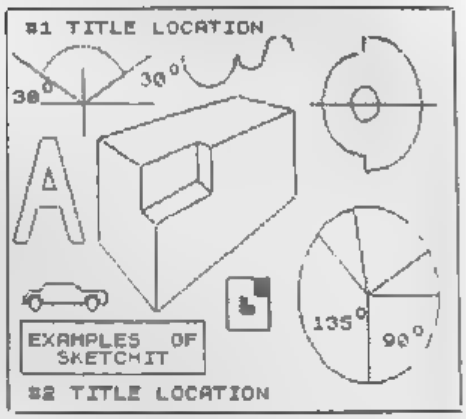


FIGURE - 2

5 - Lines where start, length, and direction are known. These are called vectors.

6 - Arcs of circles.

7 - TITLES & COPY the display. Have a printer ready.

8 - SAVE the display on tape. for future work. Have recorder ready and running.

9 - Exchange DRAW and ERASE modes.

Ø - EXIT the ON ERR mode. Gives a choice to continue or CLS and start over.

Figure 2 shows a variety of shapes and designs produced by the action of SKETCHIT-G. Let us run thru one of the preceding actions...say the arc drawing feature initiated by the "6" key. In particular, refer to the arc shown in the upper left corner. To draw an arc, we must know its radius and the center point of that radius. We must also know what sector it covers and in this program the sector is defined by the starting radius and the ending radius. In turn, these are measured counter-clockwise in degrees from an arbitrary horizontal line, the initial line. In this example, the starting radius is 30 degrees from the intitial line and the terminal radius is 150 degrees from the initial line. Of course none of the straight lines that are in this arc sample are needed to draw the arc. They are included here merely to help visualize the arc drawing parameters, and how the arc relates to them. Notice too, that the arc draws itself in a counter-clockwise direction. All of the curves in this program, circle and half-circle included, develop in this same way. All angles are measured form the same horizontal, initial line, counter-clockwise from 0 to 360 degrees. This concept is more or less customary in mathematics and related subjects.

You will note that the "3" key is for both straight lines and half-circles. The straight line is defined by its starting coordinates and "pitch". Pitch here means how far up or down and how far left or right the second terminal is from the first, measured in pixels. If you are using the "3" key for drawing half-circles, then the line is not drawn. Instead a half-circle is drawn for which the line would be its diameter. And the half-circle plots from the initial coordinates, counter-clockwise to the second terminal of such imaginary lines.

One other thing. Since the program is in BASIC and a lot of calculation is performed, you may notice that key response and INPUT prompting is slow at times. A good rule to follow: look at your INPUT data on the bottom line of the screen before pressing ENTER.

The various sub-routines in the program listing are identified by REM lines to help you to analyze it and to see what makes it work. Have fun.

Dear MSCRIPTERS:

by Jack Dohaney

When I first started customizing MSCRIPT a couple of eons ago, I sort of promised a semi-regular newsletter to keep users abreast of latest developments. After two issues it died. Why? Because suddenly there were more Mscript owners than I could cope with. I simply don't have the resources (time & money) to do mailings to so many people.

Now it appears that TIME DESIGNS may serve as an excellent vehicle for such a newsletter. It may appear sporadically, since I tend to say something only when I have something to say.

BUG REPORT (Hey, I'm only human)

Bug #1: If your MSSA won't Add Text from wafer, then BASIC Line 375 needs fixing. Should be:

```
375 NEXT I: PRINT " "; INVERSE 1;C$; INVERSE 2;" ";
    IF C$="L" OR C$="A" OR NOT U THEN RETURN
```

Bug #2: If you have an AERCO Interface and your printer messes up when printing tex, then you probably need Version 5.1. See VERSION REPORT below. Stock Mscript's AERCO output routine had sneaky flaws which hid from me up to and including version 5.0. These flaws cause the strobe generated by the AERCO Interface to sometimes be too long for many printers, with wierd results. The strobe is the signal from the interface to the printer that says: I got good data for you NOW. The strobe should last no longer than the data, but long enough for the printer to notice it: a few microseconds. With the AERCO Interface, strobe duration is controlled by software rather than hardware, and interrupts must be OFF when the strobe is initiated. If an interrupt occurs after the strobe is started, the strobe will be too long for some printers.

Another output routine flaw can let printers lacking a "print-error" signal generate a "false error signal" which aborts printing prematurely.

The problem described by Mel Rout in the "TS Communique" column on page 5 of the March/April 87 issue of TDM, indicates he needs Version 5.1 or higher. Joe Williamson's reply can probably be disregarded, with thanks for the effort.

Thanks to Jerry Chamkis of AERCO and P.E. Skipper of the world at large, for helping to defumigate this one. Actually it's two separate bugs, but I call it one. All 3 of my printers, of course, work perfectly with Version 5.0. Alas, all printers are not created equal, but all should work perfectly with V5.1 and higher.

VERSION REPORT

Version 6 is in the works. I decline to speculate as to when it will be ready, or to describe new features in advance. Now's the time to send me your suggestions. Customized Mscript users who have "registered" with me and paid their dues will be notified when V6 is available.

Versions between 5 and 6 (such as 5.2) are "intermediate" versions, steps along the way toward V6. BASIC Line 1 tells full version number. The latest version can always be obtained by request. Payment is in the form of voluntary donation, as usual.

VERSION 5.1 corrects sneaky machine-code flaws in the AERCO print routines of versions 5 and below. The pokes necessary to upgrad from V5 to V5.1 are too extensive for publication here.

VERSION 5.2 changes BASIC so that a backup may include current text. This has several uses: your backup may include your letterhead and/or usual printcode definitions. Or you can reload work-in-progress all at

once, rather than reloading first the program and then the textfile. Version 5.2 simplifies the View Memory facility, to make room in BASIC for Backup's include-text function.

There are at present four "standard varieties" of Customized Mscript: MS5T for cassette tape only, MS5A for A&J and cassette, MS5D for AERCO disc and cassette, and MS52 for ZEBRA disc and cassette. SPECIAL VARIETIES of Customized Mscript are available for use with Tasman and Byte-Back serial interfaces and for RAMEX disc, but not yet for Oliger or Larken disc. I would need those devices on loan in order to develop such varieties.

DUES REPORT

Many Customized Mscript owners have not "paid their dues", probably because they're ignorant of the big picture. The facts are these: I do NOT hold the Mscript copyright and cannot legally sell Customized Mscript, or control its distribution. Dealers who distribute Customized Mscript rather than Stock Mscript do not pay me a cent. The only compensation I get for the enormous work of customizing Mscript comes directly from happy and fairminded users in the form of voluntary donations. To date I have been compensated for my work at the rate of roughly $2 per hour. My thanks to all Customized Mscript users who have registered, and paid their dues or expressed their appreciation.

THE FUTURE

I believe that the wheel needs to be re-invented continuously. Were it not, we'd still be riding around on round rocks instead of steel belted radials. I plan to go right on improving Mscript and other things as long as possible.

Jack Dohany, 390 Rutherford Ave., Redwood City, CA 94061 (415) 367-7781. Send legal size SASE for list of available "Fairware" programs.

ZEUS UTILITY

by Richard Hurd

```
5 REM
      ZEUS SOURCE CODE TO
         MSCRIPT CMSV5
         or TASWORD II
           CONVERSION

        (C)1987 Richard Hurd

  10 REM This program uses and
is based on the mnemonics looku
ptable from the ZEUS assembler

  20 PRINT ''TAB 13,"Z2M/T"''''
Conversion routine to convert
 Zeus formatted Source Code to
  either or both:"''" Mscript (
CMSV6T)"'" TASWORD II"
  30 BEEP .1,7: PRINT ''''TAB 10
;"Please LOAD"''TAB 3,"ZEUS (Fo
rmat) Source Code"
  40 LOAD ""CODE 32768
  50 LET length=USR 61431  POKE
46927+length,255
  60 CLS   PRINT ''''" Press appr
opriate key"''" 1) MSCRIPT (CMS
V5T)"'" 2) TASWORD II"'" 3) BOT
H 1 & 2"
  61 LET i=CODE INKEY$  IF i=COD
E "" THEN  GO TO 61
  65 IF i<49 OR i>51 THEN  GO TO
 61
  70 IF i=49 OR i=51 THEN  CLS .
 PRINT '''" Saving Z2Mfile CODE
 46927,";length. SAVE "Z2Mfile"
CODE 46927,length  IF i=49 THEN
  GO TO 100
  80 LET length=USR 61556  POKE
61552,79: POKE 61553,183  POKE
61554,48: POKE 61555,117
  90 PRINT '" Saving Z2Tfile COD
E 30000,";length: SAVE "Z2Tfile
"CODE 30000,length
 100 CLS : PRINT '''TAB 4;"That'
s It!": STOP
 110 LOAD ""CODE   CLEAR 29999
RUN
 120 CLEAR   SAVE "Z_2_M/T" LINE
 110: SAVE "Z_2_M/T"CODE 61015,
610
```

The following routine converts ZEUS source code files into an ASCII file for either of the two most popular TS2068 word processors: MSCRIPT and TASWORD TWO. A hex dump is included for ease of data entry.

```
[illegible]  41 0A 41 44 43 20 08 41
[illegible]  44 44 20 08 41 46 27 0A
EE67  41 46 0A 41 4E 44 20 08
[illegible]  42 0A 42 43 0A 42 49 54
EE77  20 08 43 0A 43 41 4C 4C
EE7F  20 08 43 43 46 0A 43 50
EE87  20 08 43 50 44 0A 43 50
EE8F  44 52 0A 43 50 49 0A 43
EE97  50 49 52 0A 43 50 4C 0A
[illegible]  44 0A 44 41 41 0A 44 45
EEA7  0A 44 45 43 20 08 44 45
EEAF  46 42 20 08 44 45 48 4D
EEB7  20 08 44 45 46 53 20 08
[illegible]  44 45 48 57 20 08 44 49
EEC7  0A 44 49 53 50 20 08 44
EECF  4A 4E 5A 20 08 45 0A 45
0
EED7  49 0A 45 4E 54 0A 45 51
EEDF  55 20 08 45 58 20 08 45
EEE7  58 58 0A 48 0A 48 41 4C
EEEF  54 0A 48 4C 0A 49 0A 49
EEF7  4D 20 08 49 4E 20 08 49
EEFF  4E 43 20 08 49 4E 44 0A
EF07  49 4E 44 52 0A 49 4E 49
EF0F  0A 49 4E 49 52 0A 49 58
EF17  0A 49 59 0A 4A 50 20 08
EF1F  4A 52 20 08 4C 0A 4C 44
EF27  20 08 4C 44 44 0A 4C 44
EF2F  44 52 0A 4C 44 49 0A 4C
EF37  44 49 52 0A 4D 0A 4E 43
[illegible]  0A 4E 45 47 0A 4E 4F 50
EF47  0A 4E 56 0A 4E 5A 0A 4F
[illegible]  52 20 08 4F 52 47 20 08
0
```

Hex Dump of Z_2_M/T

Beta Basic 3.0

Over 100 New Commands For Spectrum/Emulated 2068

Reviewed by Robert D. Hartung

When I first loaded Beta Basic and began working my way through its well-written 90-page manual, my impression was, "Wow! This is almost like having a poor man's QL!" Of course this extension to Sinclair Basic must work within the hardware limitations of a T/S environment, but after a year of learning my way around it a bit I am even more convinced that it provides our "toy" computers with what is probably the most powerful Basic programming language available on any 8-bit micro today.

Many of its over 100 new and enhanced commands and functions have several variations. Except for TS2068 commands that are not available in Spectrum mode (STICK, SOUND, ON ERR, RESET, FREE) it is completely compatible with Basic programs that have been entered and saved on either a TS2068 or Spectrum. (Free bytes function MEM(), and ON ERROR line no. and ON ERROR statement statement ... are supported.) This means that any T/S Basic program that will fit under a 46679 RAMTOP may be loaded in with BB in-residence and and its toolkit features used to edit, split, join, re-number, or make block-moves or copies in the listing with all discrete line-references changed accordingly. Keying in a zero before any line number immediately brings that line into the editing area without shifting the listing display. AUTO line numbering in any step may be turned on or off.

After all this, if no BB-specific command words have been inserted into the listing, it may be saved again as a TS2068 or Spectrum program, or selected blocks of the listing or data in memory may be saved from the complete program. If saved with the special line 0 containing BB directives, it may contain any of the BB commands and will run with the BB code in-residence. As supplied, BB is cassette and MicroDrive compatible, with adaptations available that will work with WafaDrive as well as Opus, Kempston, Oliger, and other Spectrum-compatible DOSs. It takes about 125 seconds to load from tape.

Personally, I would think this program worth having for its editing features alone, but that is barely the beginning of all that it does. Probably the one most powerful feature is that procedures can be created, each of which may then be called by a single word. In effect, this allows the user to create a library of new command words, limited only by your ingenuity and available memory. Procedure parameters and internal variables designated as LOCAL do not affect other program variables which have the same names. DEFAULT values for parameters and other variables may be pre-defined. DATA may be used as a procedure parameter.

By using stored addresses, FOR-NEXT loops run at constant speed anywhere in the listing, unlike T/S Basic, and are about 2 times faster at the first line, 5 times faster at the 100th, and 17 times faster at the 500th line. GO TO and GO SUB are also faster and RETURN is just as fast from the last line as from the first. The DO-LOOP structure is supported as well as WHILE, UNTIL, EXIT IF, ON, and ELSE. SORT will re-arrange any array of strings, numbers or letters, in ascending or descending order--400 of them in about 3 seconds. INARRAY and INSTRING will search for any given target string. ITEM() checks for the last item of DATA and whether any item is numeric or a string. READ LINE allows READ to work with DATA that otherwise would need quotes.

Graphic and display control are provided in any of 127 user-defined windows, with wraparound pixel-by-pixel and attribute ROLL in any direction as well as SCROLL in any direction. FILL will surround all closed figures with designated PAPER color or fill any closed figure at given coordinates with designated INK color. CSIZE provides characters in any size from 80 columns per line to one filling the entire screen, as well as reducing or enlarging any text or any portion of the display put into a string by GET, which you may then PLOT anywhere you want it on the screen. Plotting scale and 0,0 origin coordinates may be set with four special variables.

Other commands and functions include OVER 2 which allows superimposing printed or plotted text or graphics without affecting what is already there. STRING$ will print any character or string a given number of times. Cursor control codes may be imbedded in text to change print positions in the display. USING or USING$ may be used to align printed columns of integer and decimal values and truncate to the desired number of places. LLIST expands token-words and TAB is translated to the proper number of spaces before being sent to a full-size printer. (The ProFile printer driver routine will work in BB/Spectrum mode by using POKE 63688,84 and POKE 63689,31 before moving it just below the BB code.) SCRN$ recognizes UDGs along with the normal characters recognized by SCREEN$. A real-time CLOCK provides options of display and/or audible alarm and/or timed GO SUB.

KEYWORDS n provides one-key entry of keywords, letter-by-letter entry, or combination of both. The KEYIN command actually can be used to create self-writing programs. All variables and their contents may be listed, or all lines that contain a given reference or procedure may be displayed in succession for examination or editing. The ALTER __ TO __ command may be used (but with caution!) to change every occurance of a given word or

character-sequence wherever it appears in memory. A TRACE function may be used to single-step through a program for de-bugging. Strings, arrays, or blocks of data may be displayed for editing and may be moved, joined, or copied in memory. Any part of memory may be put into a string. LIST a=x,b=y,c$="z" is supported.

Any or all character keys may be user-defined to produce a complete program line or a conversion formula or an entire subroutine with one key-stroke; and these definitions may be saved with the program. Hex and decimal conversion functions are provided; as is decimal to binary and bit-by-bit AND, OR, and XOR of two numbers. CLEAR nn768 moves RAMTOP without loss of variables. BREAK will stop machine code endless loops. Listing formats indent wraparound lines to the fifth column for easier reading. Options provide listing of multi-statements on separate lines, with some commands indented further.

A random number function is 2 1/2 times faster than RND#n. SINE and COSE functions give four-place accuracy 6 times faster than SIN and COS. DPEEK yields decimal result of a double PEEK and DPOKE does LSB, MSB double POKE of a decimal number to a given address. CHAR$ converts 0-65535 integer values to two-character strings and NUMBER converts them back, at a saving of 60% in use of memory. MOD gives remainder of one number modulo another.

Since no review can provide the "picture that is better than a thousand words," with the permission of BetaSoft I have written a 20K-byte demo which shows typical listings and runs actual examples of about 85 of the new and enhanced commands of this remarkable software for the Spectrum/Emulated TS2068. It is available for $5.00 to cover the cost of postage, packaging, tape, and handling by writing to me at 2416 N. County Line Rd., Huntertown, IN 46748. Please note that, to protect BetaSoft's copyrights, the version of Beta Basic included to drive this demo will NOT allow editing or use with any other listing but the demo may be used as a tutorial when loaded with normal BB. Beta Basic 3.0 may be purchased for 15.50 in British pounds (about $25.00) from BetaSoft, 92 Oxford Rd., Moseley, Birmingham B13 9SQ, England. Payment may be by international money order, or MasterCard for easier currency exchange.

Next Issue:

Professional TASWORD TWO.

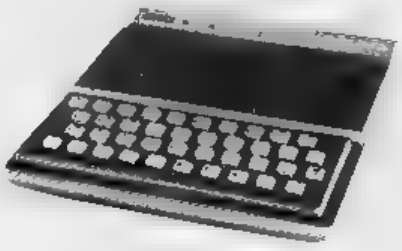
STARFLEET 2068.

"Beyond the User Manual"

New series for the ZX81/TS1000/TS1500.

Programming Concepts

by Albert F. Rodriguez



PROGRAMMING CONCEPTS has been a multi-part feature in TIME DESIGNS, and deals with programming on the Sinclair ZX81 (or TS1000/TS1500). The chief example used in this series is a game program called "ZX81 TIC-TAC-TOE" which has been discussed extensively. In this installmant we feature the actual BASIC listing of the program.

If readers do not wish to type in the listing, there is a cassette tape available (non-listable) for $12 from the author: Albert F. Rodriguez, 1605 Pennsylvania Ave., #204, Miami Beach, FL 33139. Also, you can get a print-out of this listing along with the array content and program declarations for $6.20 ppd (foreign buyers add $2.00 for the cassette, or $1.00 for the listing).

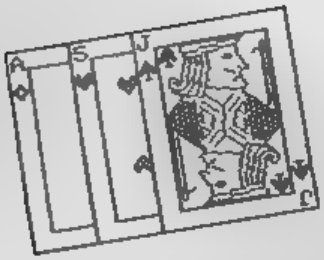
In the next issue, we will have a complete line explanation of the program and more.

ZX TIC TAC TOE

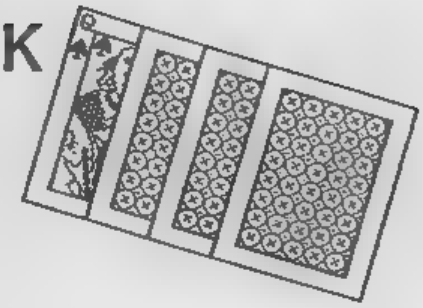
```
 [illegible]REM  TTT
 [illegible]SAVE  TTT
    [illegible]=
      [illegible]
      [illegible]$
-3  LET [illegible]
4   LET [illegible]$  = P$
15  [illegible]
18  FOR  [illegible]
19  LET  [illegible]
20  NEXT [illegible]
25  LET  TP=0
30  LET  P$= [illegible]
[illegible]  [illegible] MAK
[illegible]  [illegible] B=1
[illegible]  [illegible] P=
[illegible]  [illegible]
[illegible]  [illegible] P[illegible]
[illegible]  [illegible] B
[illegible]  [illegible] GAME
100 PRINT AT [illegible]  TP  TI  IN
100e PRINT AT  [illegible] M[illegible] AT  ME
OMPUTER
1003 PRINT AT [illegible] AT  TIC-TA-
TOE
1004 PAUSE 150
1005 [illegible]
1006 PRINT AT  12  REMEMBER
1007 PRINT AT  8  TO STRETCH
GAME
1008 PRINT AT  9  DURING  OF  T
ION
1009 PRINT AT  8,  PRESS  SPACE
KE
1010 PRINT AT  10  TO RESTART
1011 PRINT AT 12 10  T HE  OD
1012 PRINT AT  [illegible] 5  THEN  PRESS
ENTER  KE
1013 PRINT AT  [illegible] 3  NE TH EN TO
RUN  OR  CLEAR
1014 PRINT AT  4 16  IF  DO
1015 PRINT AT  6 5  OU  THEN  H
AVE TO RELOAD
1016 PRINT AT  8 5  CAME FROM CA
SETTE
1017 PAUSE 720
1018 CLS
1019 RETURN
2001 LET L=0
2002 FOR L=1  4
2003 PRINT TAB  I  TAB  I
2004 NEXT
2005 LET C=
2006 IF C=3 THEN GOTO 2016
2007 FOR I=1  16
2009 PRINT I$
2010 NEXT I
2011 GOTO 2002
2012 RET RN
3001 LET D=2
3002 FOR J=1 TO 9
3003 IF  =4  THEN LET D=D+
3004 IF  =7  THEN LET D=D+
3005 PRINT AT  E  $
3006 LET  =
3010 PRINT AT  F  $
3011 LET  =
3012 PRINT AT  G  $
3014 NEXT
3015 RETURN
4003 PRINT AT 0 22  ZX81
4004 PRINT AT  19  TIC-TAC-TOE
4005 PRINT AT 2 20  TO
4006 PRINT AT  19  A. RODRIGUEZ
4007 PRINT AT 17 1  PLA BY THE
```

Continued Next Page.

```
4003 PRINT [illegible]
ES ▩
4009 PRINT AT [illegible] PLA ER [illegible]
[illegible]
4010 RETURN
5000 INPUT K$
5001 PRINT [illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
5012 PRINT [illegible]
AGAIN
5014 PAUSE 120
5015 PRINT AT 20,0

5016 PRINT AT 21,0

50[illegible] GOTO 5000
6000 LET K=0
6001 LET [illegible]
6002 FOR [illegible]
6003 LET [illegible]
6004 LET [illegible]
6005 IF K$=J$ THEN LET [illegible]
+1
6006 IF K$=J$ THEN LET [illegible]
6007 NEXT [illegible]
6008 IF K>=2 THEN GOTO 6013
6009 RETURN
6013 PRINT AT 21,1,"REPEAT  TRY
AGAIN
6014 PAUSE 120
6015 PRINT AT 20,1
6016 PRINT AT 21,1,

601[illegible] GOSUB [illegible]
6018 GOTO 6001
7001 FOR J=1 TO 9
7002 IF K$=C$[illegible] THEN LET [illegible]
▩
7003 NEXT J
7005 GOSUB IN
7006 IF F$=[illegible] THEN RETURN
7007 GOSUB PKM
7008 GOSUB OUT
7009 IF F$="WIN" THEN RETURN
7010 GOSUB SET[illegible]
7011 GOTO 7001
7012 GOSUB PST
7013 GOSUB RCD
7014 GOSUB CHK
7015 RETURN
7016 GOSUB ASG[illegible]
7017 GOSUB [illegible]
7018 GOSUB CHK
7019 RETURN
7020 FOR [illegible]=1 TO 9
7021 IF K$=C$[illegible] THEN LET C$[illegible] =
▩
7022 NEXT [illegible]
7024 GOSUB PSTN
7025 RETURN
7026 FOR P=1 [illegible]
7027 LET [illegible]
7028 NEXT P
7029 LET [illegible]
7030 LET P=1
7032 FOR [illegible]=1 [illegible] 9
7033 IF [illegible]=4 OR [illegible]=7 THEN [illegible] P=P+
1
7034 LET L(P)=L(P)+CODE [illegible]
7035 NEXT [illegible]
7036 LET P=P+[illegible]
7037 FOR [illegible] STEP [illegible]
7038 LET [illegible]
7039 IF [illegible] OR [illegible]=8 THEN LET [illegible] P+
1
7040 IF [illegible] =8 THEN LET [illegible]
8
7041 NEXT [illegible]
7042 LET P=P+1
7043 FOR S=1 [illegible]
7044 LET [illegible]
7045 NEXT S
7046 LET P=P+[illegible]
7047 FOR [illegible]=3 [illegible] STEP [illegible]
7048 LET [illegible]
7049 NEXT [illegible]
7050 FOR R=1 [illegible]
7051 LET TL=TL+[illegible]
7052 NEXT R
7053 RETURN
7054 LET CTR=CTR+1
7055 LET LN=7069
7056 LET U=0
705[illegible] FOR R=1 TO 8
705[illegible] IF R[illegible]2 THEN LET [illegible]
7059 IF R[illegible]3 THEN LET [illegible]+2
7060 IF [illegible] THEN LET [illegible]
7061 IF [illegible] THEN LET [illegible]
7062 NEXT R
7063 IF [illegible] THEN [illegible]
7064 IF CTR=[illegible] THEN [illegible]
7065 RETURN
[illegible] UB [illegible]
[illegible] LET F$=[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible] II
[illegible] III
[illegible]
7100 FOR [illegible]=1 TO 215
7101 IF TL=T[illegible] THEN LET [illegible]
7102 IF TL=T[illegible] THEN LET [illegible]
710[illegible] NEXT [illegible]
7104 FOR J=1 TO 22
7105 LET I=1
710[illegible] IF TL=[illegible] AND C$(I)=A$[illegible]
THEN LET [illegible]
710[illegible] IF TL=[illegible] THEN LET J=22
7108 NEXT [illegible]
7109 FOR J=1 TO 13
7110 LET I=2
7111 IF TL=B[illegible] AND C$(I)=A$[illegible]
THEN LET K$=D$[illegible]
7112 IF TL=B[illegible] THEN LET [illegible]
7113 NEXT [illegible]
711[illegible] FOR [illegible] 16
711[illegible] LET I=3
7116 IF TL=B(J) AND C$(I)=D$(J)
THEN LET K$=E$(J)
7117 IF TL=B[illegible] AND C$(I)=D$(J)
THEN LET J=[illegible]
7118 NEXT J
7119 IF TL=3804 AND C$(4)='4' THEN LET K$='4'
7120 IF TL=2904 AND C$(4)=▩' THEN LET K$="7
7121 IF TL=2754 AND C$(4)='▩ THEN LET K$="7
7122 FOR J=1 TO 15
7123 LET I=7
7124 IF TL=F[illegible] AND C$(I)[illegible]
THEN LET K$=H$[illegible]
712[illegible] IF TL=F(J) AND C$(I)[illegible]
THEN LET [illegible]=1[illegible]
7126 NEXT J
712[illegible] IF TL=2[illegible] AND C$(6)[illegible]▩ THEN LET K$= 7
712[illegible] IF TL=2758 AND C$(5)=▩ THEN LET K$= 4
7129 IF TL=2[illegible] AND C$(6)=▩ THEN LET K$= 4
7130 IF TL=3[illegible] AND C$(3)[illegible]▩ THEN LET K$= 2
713[illegible] IF TL=3[illegible] AND C$(9)='▩ THEN LET K$= 1
7132 IF TL=2913 AND C$(9)=▩ THEN LET K$= 3
713[illegible] IF TL=3813 AND CODE C$(3)+CODE C$(6)=223 THEN LET K$= 4
7134 IF TL=3813 AND CODE C$(2)+CODE C$(3)=219 THEN LET K$='8
7135 IF TL=3813 AND CODE C$(2)+CODE C$(3)=360 THEN LET K$='6'
7136 IF TL=3813 AND CODE C$(8)+CODE C$(9)=223 THEN LET K$='4
713[illegible] IF TL=38[illegible] AND CODE [illegible]+CODE C$[illegible]=37[illegible] THEN LET K$='9
7138 IF TL=2899 AND CODE C$(1)+CODE C$(4)=375 THEN LET K$=[illegible]
7139 IF TL=3[illegible] AND CODE C$[illegible]+CODE C$[illegible] THEN LET K$[illegible]
7140 IF [illegible] AND CODE C$[illegible]+CODE C$(8)=360 THEN LET K$[illegible]
7141 IF TL=[illegible] AND CODE C$[illegible]+CODE C$(4)=360 THEN LET K$[illegible]
714[illegible] IF TL=290[illegible] AND CODE C$[illegible]+CODE C$(9)=360 THEN LET K$=[illegible]
7143 IF TL=3[illegible] AND CODE C$(5)+CODE C$(8)=360 THEN LET K$=[illegible]
7144 IF TL=2900 AND C$(4)=▩ THEN LET K$= 6
7151 IF TL=3813 AND CODE C$(4)+CODE C$(7)=221 THEN LET K$=[illegible]
715[illegible] IF TL=3813 AND CODE C$[illegible]+CODE C$(2)=219 THEN LET K$=[illegible]
7154 IF TL 3663 AND C$(9)=[illegible] THEN LET K$= 2
714[illegible] LET [illegible]
[illegible]
[illegible]
[illegible]
900[illegible]
900[illegible] PRINT [illegible]
[illegible]
900[illegible] GOSUB [illegible]
900[illegible] GOSUB [illegible]
900[illegible] RETURN
9003 GOSUB DSPL
9004 LET B=[illegible]
9005 GOSUB PL[illegible]
9006 GOSUB [illegible]
9007 RETURN
9008 GOSUB D[illegible]
9009 LET B=25
9010 GOTO 9005
9011 GOSUB DSPL
9012 LET B=23
9013 GOTO 9005
9014 LET A=0
9015 FOR I=1 TO 25
9016 IF I=1 THEN PLOT A B
9017 LET A=A+1
9018 PLOT A B
9019 IF I=3 OR I=11 OR I=2[illegible] THEN LET A=A+2
9020 NEXT I
9021 RETURN
9022 GOSUB D[illegible]
9023 LET A=4
9024 GOSUB PL[illegible]
9025 GOSUB [illegible]
9026 RETURN
902[illegible] GOSUB DSPL
9028 LET A=14
9029 GOTO 9024
9030 GOSUB DSPL
9031 LET A=2[illegible]
9032 GOTO 9024
9033 LET B=42
9034 FOR I=1 TO 11
9035 IF I=1 THEN PLOT A B
9036 LET B=B+1
9037 PLOT A,B
9038 IF I=11 THEN GOTO 9042
9039 IF I=2 OR I=6 OR I=9 THEN LET B B-2
9040 LET B=B-3
9041 PLOT A B
9042 NEXT I
9043 RETURN
9044 GOSUB DSPL
9045 LET A=1
9046 LET B=42
9047 FOR I=1 TO 12
9048 IF I=1 THEN PLOT A,B
9049 LET A=A-1
9050 LET B=B+1
9051 PLOT A,B
9052 IF I=12 THEN GOTO 9058
9053 IF I=2 OR I=8 THEN LET A=A+
2
9054 IF I=2 OR I=6 THEN LET B=B+
2
9055 LET A=A+3
9056 LET B=B-3
9057 PLOT A,B
9058 NEXT I
9059 GOSUB WIN
9060 RETURN
9061 GOSUB DSPL
9062 LET A=2[illegible]
9063 LET B=4[illegible]
9064 FOR I=1 TO 13
9065 IF I=1 THEN PLOT A B
9066 LET A=A+1
9067 LET B=B-1
9068 PLOT A,B
9069 IF I=13 THEN GOTO 9075
9070 IF I=7 THEN LET A=A-2
9071 IF I=7 THEN LET B B-[illegible]
9072 LET A=A-3
9073 LET B=B-[illegible]
9074 PLOT A,B
9075 NEXT I
9076 GOSUB WIN
9077 RETURN
9078 PRINT AT 18,1
     AT 20 1,"

9080 RETURN
9081 PRINT AT 16,6," "COMPUTER H
AS WON"
9082 PAUSE 900
9083 CLS
9084 RETURN
9085 GOSUB DSPL
9086 PRINT AT 16,6,"' "GAME IS A
DRAW
9087 PAUSE 900
9088 CLS
9089 RETURN
```



High-Res BLACKJACK
For the ZX81/TS1000/TS1500



by Fred Nachbaur

For many years now, we ZX81/TS1000/TS1500 users have had to contend with VVLR (Very Very Low Resolution) graphics. Not any more! With nothing more than a suitable static memory, mapped from 8-16k, we now have true high-resolution graphics for our machines!

By "suitable" I only mean that the static RAM's data must be accessible during refresh time. If you have a HUNTER Board, you can make a very simple modification to meet this requirement. Simply cut the line between the RAM chips pins 20 and edge connector pin 16A (RD*); bridge the resulting gap with a 4.7k resistor. Now connect a diode with its cathode (banded) end to edge connector 23A (RPSH*), and the other end to the RAM side of the new resistor. Alternately, contact SILICON MNT. COMPUTERS about our "SCRAM" board, which will do the job with no modifications.

Silicon Mountain also has "SCRAM HI*RES EXTENDED BASIC", which gives you 38 new high-resolution commands from BASIC. The prupose of this series of articles is to give a tutorial on the use of SRAM HI*RES, demonstrating how these new commands can be used to write spectacular graphics software...entirely from BASIC.

Editor's Note: You can contact Silicon Mountain Computers by writing to: C-12, Mtn. Stn. Group Box, Nelson, B.C., V1L 5P1, Canada. A complete line of interesting products are available for the Sinclair ZX81, TS1000 and TS1500. The "SCRAM" NVM Board is priced at $39.95 postage paid; the "SCRAM HI*RES EXTENDED BASIC" software package is priced at $24.95 ppd.; and other high-resolution graphics programs are available including the games DUNGEON OF YMIR and HIGH-RES CHESS.

Since this article presupposes that you have SRAM HI*RES EXTENDED BASIC, I can already hear charges of "vested interest" in publishing a program that makes use of it. I must plead "no contest", but also point out that it is common practise for manufacturers in other fields to provide after-sale support via their trade magazines (like TIME DESIGNS). Granted, the SRAM HI*RES manual gives all the basic info on this new operating system, much as your computer manual gives "all you need to know" about your ROM operating system. Still, many users will find it necessary to get additional explanation in the way of articles and sample listings. This article, and the accompanying BLACKJACK listing, will attempt to get you familiar with the use of some of the new SRAM HI*RES commands.

THE DOCUMENTATION

Before we get on with it, here is the documentation for the BLACKJACK program published here. Look over the artwork for this article for an idea of the displays you will be seeing; these are all ACTUAL screen dumps of in-progress games!

The game of BLACKJACK is one of the most popular casino games, and for a good reason; it is one of the very few casino games where you can actually have odds over the house. A skilled Blackjack player can actually (in the long run) come out ahead, because of an estimated 2% advantage. Compare this to horse racing, with an estimated 25% loss, or lotteries with a typical 50% loss!

GETTING STARTED

Start by hooking up your ZX81, TS1000 or TS1500 (with 8k static RAM and a Timex 16k RAM pack). Note #1: Larger RAMs will work, but you will still need the static RAM in the 8-16k region. Many 64k RAMs WILL NOT work on the 1500, however. Note #2: If you're using a TS1500, you'll still need the Timex 16k pack in addition to the internal 16k. This is for the hardware reasons unique to the TS1500.

The first thing we'll do is enter a program that actually will be run only once! This is because there isn't enough room in 16k if we include it in the main listing. The purpose of this is to set up a dimensioned string array to hold the character codes for the UDG's used to print the images of the cards. Incidentally, this little "trick" demonstrates a use for your static RAM even in non-high res programs; use it anytime you need to import or export data (variables) between programs.

Enter LISTING 1. You don't have to enter the REM lines, they are only provided to help with the graphic characters. Proof the listing, save it to tape just in case you "mess up" later, and run it. This dimensions C$(4,13,7,5), then proceeds to fill it. The characters it is filled with correspond to the UDG (User Defined Graphic) characters that will be set up in the main program. Note the significance of the dimension numbers: "4" (number of suits), "13" (number of cards in each suit), "7" (number of lines in each card image) and "5" (number of columns). Once the array is filled, the contents of the array is POKEd into your static RAM.

Now enter NEW, LOAD the SRAM HI*RES core program, and delete the extraneous lines 3-9999. Now type in line 2 of LISTING 2. Note that this also deletes the 64-column PRINT driver routine, which we won't be needing in this program. Just for safety's sake, enter the following POKEs, which prevent accidentally calling the deleted 64 column routine:

```
POKE 18080,208
POKE 18081,75
```

Then, LIST 2 and POKE 16419,2. This prevents your listing from getting stuck at line 0. Henceforth, using only LIST (number greater than 2) instead of LIST (only) then, enter the program lines starting at 9900.

Now comes the nice part. Enter RUN 9900. The C$ array will be re-created, and filled with the proper contents from your 8k static RAM board! While we're at it, we define a few of the important variables that won't change throughout the program. You can now delete lines 9900 onwards, leaving you only with the line 2 REM and of course the line 6000 REM. Henceforth, NEVER USE RUN OR CLEAR, or you will destroy these variables!

Though this seems restrictive, look at how much memory we have saved with this approach; the entire contents of LISTING 1!!

ENTERING THE PROGRAM

Now we're finally set up to enter the program. For your convenience, we're publishing the entire program in this issue, reserving the detailed tutorials for subsequent issues. This will give you an actual, running program, even if you may not understand how it all works.

I suggest using FAST mode for program entry. The most laborious aspect is entering all the UDG and sprite definitions starting at line 9000, so I suggest you do this first to get it out of the way. If this seems like a lot of work, consider this: you don't have to design the pictures, break them into individual UDG's, translate them to hex, debug, modify, and debug again. I already did all that. All you have to do is type in the numbers, proofing and correcting as required.

I should note that the separator characters (or commas) between the hex numbers can be anything you wish; I like commas, but you may prefer spaces, or whatever. Similarly, I used other characters between each UDG (group of 8 numbers) to aid in entering the numbers originally; this merely indicates which character the subsequent UDG defifiniton relates to, and made it easier for me to perfect the images. They have no other significance. Again, you might prefer to replace these with spaces, asterisks, or whatever.

Remember, once you start entering the program you must NEVER use RUN or CLEAR. To guard against disaster, save to tape frequently. A few minutes spent saving, while you go get another cup of coffee or stretch your legs, will be well worth it if your kids stumble over your power cord or your cat jumps onto the RAM pack. After entering each group of lines (e.g. every 100) proof your work. Errors will be easier to spot if you are dealing only with a small group of lines. Be careful to type in the listing exactly; at some points, even an omitted semicolon or space will cause improper program operation.

When you're all done, and have saved to tape, get ready for the show. Make a final save to tape using GOTO 8900; this will then autorun after the SAVE (or subsequent LOAD). The program initializes all the UDGs and sprites at line 9000, jumping to the actual program start at line 2000 when complete. (Though it looks like it would take a long time to execute all those definitions, i.e., 128 UDGs and 26 sprites, it actually happens in less than a second!)

Next time we'll dissect the program, explaining why some of those weird commands are there and how they work. Meanwhile, have fun! Don't spend all your money in one casino!

BLACKJACK

This program simulates the casino game very closely even though it doesn't allow refinements like doubling bets or splitting aces, but it does have the capability of using multiple decks. Also, it deals from a "true" deck, making it possible for you to practice "counting cards", a skill which will dramatically improve your odds in an actual game. (Be forewarned however, if you do count cards...do it discreetly since a casino will bounce you if it even suspects you of counting!) Also on that note, I must add the following disclaimer: "High-resolution BLACKJACK is intended for amusement purposes only. It is NOT to be used for actual gambling."

On loading, the program will start by itself and ask for your "bankroll". This is how much money you have when you walk into the casino. Enter the amount you're comfortable with ($100 if you're conservative, $1000 if you're a high-roller, etc.). Then enter how many decks you wish the dealer to use. This can be one-deck (makes it easier to count cards, keeping track of how many face cards are still in the deck), or more. According to reports, most casinos use three to five.

After shuffling the cards, the dealer asks how much you wish to bet. The "house limit" is $2000, though you can of course change this by modifying the program. The dealer now deals two cards to himself and to you. His first card is face-up, subsequent cards will be face-down. Your cards will, of course, all be face-up.

Presuming that neither of you got Blackjack (ace and a 10-valued card), you are now asked "HIT?". If you want another card, press "Y". If you wish to stay, press "N". The idea is to get your hand as close to 21 as possible, without going over ("busting"). Face cards count 10, aces count as either 1 or 11. All others count as their face value.

The dealer's rules follow the convention used in most casinos. If he has less than 17, he must take another card. If he has 17 or more, he must stay. The exception is a "soft 17" (ace plus cards totalling 6), in which case he must take another card. After each hand, the dealer reveals his cards. Press P to play the next hand. (If you don't want to wait until all the cards have been turned up, simply hold down "P" and the program will go on to the next hand between cards.)

If you get Blackjack, the house pays 2:1. (This is actually a little generous, but I'm a generous kind of guy...) On the other hand, it's obviously impossible for your computer to "pay off" if you do well, so the game will continue to play until you blow your wad, at which time it will insult you. (Isn't that a lot like life?) At the end of the game, you can press "P" to play again from the beginning, or "Q" to quit to the normal operating system.

LISTING 1 CARD PATTERN GENERATOR/DOWNLOADER

```
 5 FAST
10 DIM C$(4,13,8,5)
20 FOR S=1 TO 4
30 LET U$=CHR$ (174+S)
40 LET D$=CHR$ (178+
50 FOR C=1 TO 13
60 FOR B=0 TO 100*C
70 NEXT C
80 NEXT S
90 GOTO 2000
100 LET C$(S,C,4,3)=U$
110 RETURN
200 LET C$(S,C,2,3)=U$
210 LET C$(S,C,7,3)=D$
220 RETURN
300 LET C$(S,C,2,3)=U$
310 LET C$(S,C,4,3)=L$
320 LET C$(S,C,6,3)=D$
330 RETURN
400 LET C$(S,C,2,2)=L$
410 LET C$(S,C,2,4)=U$
420 LET C$(S,C,7,2)=D$
430 LET C$(S,C,7,4)=D$
440 RETURN
500 LET C$(S,C,2,2)=U$
510 LET C$(S,C,2,4)=U$
520 LET C$(S,C,4,3)=L$
530 LET C$(S,C,6,2)=D$
540 LET C$(S,C,6,4)=D$
550 RETURN
600 LET C$(S,C,2,2)=U$
610 LET C$(S,C,2,4)=U$
620 LET C$(S,C,4,2)=U$
630 LET C$(S,C,4,4)=U$
640 LET C$(S,C,6,2)=D$
650 LET C$(S,C,6,4)=D$
660 RETURN
700 LET C$(S,C,2,2)=U$
710 LET C$(S,C,2,4)=■$
720 LET C$(S,C,4,3)=J$
730 LET C$(S,C,■,2)=D$
740 LET C$(S,C,6,4)=D$
750 LET C$(S,C,8,2)=D$
760 LET C$(S,C,8,4)=D$
```

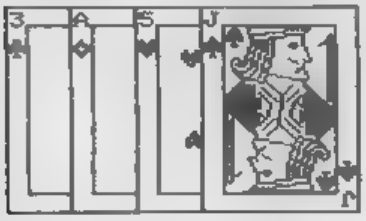
Actual Screen Samples From "BLACKJACK" ➔

```
770 RETURN
800 LET C$(S,C,2,2)=U$
810 LET C$(S,C,2,4)=U$
820 LET C$(S,C,3,3)=U$
830 LET C$(S,C,4,2)=U$
840 LET C$(S,C,4,4)=U$
850 LET C$(S,C,5,3)=D$
860 LET C$(S,C,6,2)=D$
870 LET C$(S,C,6,4)=D$
880 RETURN
900 LET C$(S,C,2,2)=U$
910 LET C$(S,C,2,4)=U$
920 LET C$(S,C,4,2)=U$
930 LET C$(S,C,4,4 =U$
940 LET C$(S,C,5,3)=L$
950 LET C$(S,C,6,2)=D$
960 LET C$(S,C,6,4)=D$
970 LET C$(S,C,8 2)=D$
980 LET C$ S,C,8 4,=D$
990 RETURN
1000 LET C$(S,C,2 2) U$
1010 LET C$(S C 2 4)=U$
1020 LET C$ S C,3 3) U$
1030 LET C$ S,C,4,2)=L$
1040 LET C$(S,C,4,4)=U$
1050 LET C$(S,C,6,2)=D$
1060 LET C$(S,C,6 4 =D$
1070 LET C$(S C 7 3 D$
1080 LET C$ S C 8 2)=D$
1090 LET C$(S C 8 4 =D$
1095 RETURN
1100 LET C$(S,C,1)=U$+"
1101 REM INVERSE $ 7 (

1110 LET C$(S,C 2)=
1111 REM INVERSE ) < = +

1120 LET C$(S,C,3)=
1121 REM IN ERSE * / +

1130 LET C$(S C 4 P
1131 REM P , INVERSE . 0 1

1140 LET C$(S,C S =
1141 REM INVERSE 2 3 4 5, THEN 7

1150 LET C$(S C,6)=
1151 REM INVERSE 6 7 8 9 A

1160 LET C$(S,C,7)="
1161 REM INVERSE B C D E

1170 LET C$ S C 8 +D$
1171 REM INVERSE F G H I

1180 RETURN
1200 LET C$(S,C,1)=U$+"BCDE"
1210 LET C$(S,C,2) FGHIJ"
1220 LET C$(S,C,3)="KLMNO"
1230 LET C$(S,C,4)="PQRST"
1240 LET C$(S,C,5)="UVWXY"
1250 LET C$(S,C,6)="Z
1251 REM Z INV SPACE GRAPHICS
1260 LET C$(S,C,7)=
1261 REM GRAPHIC R S T 3 H

1270 LET C$(S,C,8 =
1271 REM GRAPHIC G F, INV ' £

1280 RETURN
1300 LET C$ S,C,1)=U$+
1301 REM GRAPHIC 1 2 7 4

1310 LET C$(S,C,2)=
1311 REM GRAPHIC 5 T E R D

1320 LET C$(S,C,3)="+CHR$ 11+
£$'
1321 REM GRAPHIC S

1330 LET C$(S C,4)="?()><"
1340 LET C$ S C 5)=' =+-*/'
1350 LET C$(S,C,6)=',,.01'
1360 LET C$(S,C,7,='23456'
1370 LET C$(S,C,8)= 789A'+D$
1380 RETURN
1998 REM
1999 REM STORE CARDS IN SRAM
2000 PRINT CARD ARRAY FILLED. B
E SURE THAT YOUR 8-16K SRAM IS E
NABLED BE- FORE PRESSING ENTER
TO STORE.'
2010 PAUSE 4E4
2020 CLS
2030 LET DEST=8192
2040 FOR S=1 TO 4
2050 FOR C=1 TO 13

2060 FOR R=1 TO 8
2070 FOR L=1 TO 5
2080 POKE DEST,CODE C$(S,C,R,L
2085 LET DEST=DEST+1
2090 NEXT L
2100 NEXT R
2110 NEXT C
2120 NEXT S
2130 PRINT CARDS STORED. LOAD M
AIN PROGRAM,THEN GOTO 9900 TO UP
LOAD.'
2140 STOP
9000 SAVE CARD
9010 RUN
```

LISTING 2 HIGH RES BLACKJACK

```
  2 REM SPEM HIRES v2.04
  3 REM BLACKJACK
  5 REM GOTO 9000 TO START
  6 REM
 10 REM PRINT
 11 LET Q 24+R 16
 12 LET L=96+T+72
 15 IF USR HR THEN PRINT      AT
U,Q/4 E$ AT L-72 Q 4 E$
 20 IF NOT T AND D N1 N1 THEN
GOTO 100
 30 FOR L=N1 TO 8
 35 IF USR HR THEN PRINT TAB Q/
4,' ".C$ H,I,L)
 40 NEXT L
 45 IF USR HR THEN PRINT AT U-1
2,Q 4+12 CHR$ (176+H), AT U-60 Q/
4,CHR$ (174+H)
 50 IF USR HR THEN LPRINT AP I
0,262-U,SP I+13 Q+48 192-U
 60 IF USR HR THEN LPRINT R 16+
(NOT T)+96 191-96+T Q-1 Q+56 P,2
3+(NOT T ±96,184 96+T,Q+7,Q+48
 70 RETURN
 100 FOR L=N1 TO 8
 110 IF USR HR THEN PRINT TAB Q/
4,"
 120 NEXT L
 130 GOTO 60
```

$2140 PRESS P TO PLAY



26

19

* HIGH-RES *
BLACKJACK
BY F.NACHBAUR

OK,BAD BOB.
LETS SEE YOUR BANKROLL. $1000
PLAY WITH HOW MANY DECKS?

```
200 REM SET
210 LET R=C [E B$(J)-N1
220 LET J=C+N1
230 LET M INT (R/15)+N1
240 LET I R-(M-N1)*13+N1
250 LET C T+N1)=C T+N1 +N1
260 IF NOT T THEN LET A(D(N1),N
1)=M
270 IF NOT T THEN LET A(D(N1),2
=I
280 LET V I
290 IF J 10 THEN LET J=10
300 LET [ +S =D T+S +[
310 IF I=N1 THEN LET D(T+J N1
320 LET A=D(T+N1
330 GOTO 10
400 IF USR HR THEN PRINT AT 84
64-2+LEN Z$ ,Z$
410 IF USR HR THEN PAUSE AL 9
0
420 RETURN
500 IF USR HR THEN PRINT AT 8
4,N0 E$ E$ E$ E$ AT 84,N0

510 RETURN
700 REM
710 FAST
720 LET N=52*ND
730 DIM B$(N)
740 FOR C=N1 TO N
750 FOR L=C TO N STEP 52
760 LET B$ L =CHR$ C
770 NEXT L
780 NEXT C
800 RAND
810 FAST
820 CLS
830 PRINT PLEASE WAIT WHILE I
SHUFFLE.
840 PAUSE JAL 120
850 FOR C=N1 TO N
860 LET L$=B$(C
870 LET Q=INT RND*N.+N1
880 LET M$=B$ Q
890 LET B$(Q)=L$
900 LET B$(C)=M$
910 NEXT C
920 IF USR HR THEN CLEAR
930 LET =N1
940 RETURN
2000 IF USR HR THEN CLEAR
2010 IF USR HR THEN CLS
2040 IF USR HR THEN PRINT AT JAL
80 JAL 20 * HIGH-RES * A
T JAL 90' JAL 22 J$ AT JAL 1
00 JAL '20' 'BY F,NACHBAUR'
2050 FOR A=N1 TO JAL 12 STEP V
AL "3"
2060 IF SR HR THEN LPRINT A 115
+A,60-A 166+A,70-A
2070 NEXT A
2080 IF USR HR THEN PRINT AT
16,30 ' TAB 30 TAB 30,
AT 41,30, TAB 28
2090 IF USR HR THEN PRINT AT V
AL 5 JAL 46 , HI ,TAB VAL 44
' THERE. ,TAB VAL 42 IM YOJR
' TAB JAL '42' 'DEALER ,TAB VAL
42 , VEGAS TAB VAL 46' 80
B '
2100 IF USR HR THEN LPRINT C 196
,151,30,D,166,170,140,168,D,169,
175,XD 168,169
2110 IF USR HR THEN PRINT AT VAL
'130' AL '14' WHAT IS YOUR NA
ME?
2120 INPUT N$
2130 IF USR HR THEN PRINT AT 130
,N0, OK, ,N$, .' E$,E$,E$,AT 13
8,N0 LET'S SEE YOUR BANKROLL. $'

2140 INPUT O
2150 LET O=ABS INT O
2160 IF USR HR THEN PRINT O AT V
AL 150',N0,'PLAY WITH HOW MANY
DECKS?'
2180 INPUT ND
2190 GOSUB VAL '700
2200 IF USR HR THEN CLS
2205 GOSUB 500
2210 IF N=O 14 THEN GOSUB VAL 8
00"
2215 IF USR HR THEN PRINT * ,O
" * BET= ,
2220 INPUT P
2230 LET P=ABS P
2240 IF P<=O AND P<=2000 THEN GO
TO 2300
2250 GOSUB 500
2260 IF P>2000 THEN IF USR HR TH
EN PRINT LIMIT=$2000
2270 IF USR HR THEN PRINT BET T
OO HIGH"
2280 GOSUB VAL "410"
2290 GOTO VAL "2205"
2300 IF USR HR THEN PRINT '$',P,
AT 130,56,"",TAB 56,"",
,,AT 24,58,"",TAB 58,"",TAB
58 ,TAB 56,""
2310 DIM D(6)
2315 DIM A(9,2)
2320 LET E=N1
2330 LET D=E
2340 LET T E
2350 FOR X=E TO 4
2360 GOSUB 200
2370 LET T=NOT T
2380 NEXT X
2390 IF D(4)+D(6 =11 THEN GOTO 2
900
2395 GOTO 2440
2400 IF USR HR THEN PRINT AT 84,
56 'STAY'
2410 IF NOT E THEN GOTO 2580
2420 LET T=N0
2430 GOSUB 200
2440 IF D(5)>15 THEN GOTO 2500
2450 IF D(3)>N0 THEN GOTO 2470
2460 GOTO 2580
2470 IF D(5)>11 OR D(5) 8 THEN G
OTO 2580
2480 LET D(5)=D 5,+10
2490 LET D(3)=-N1
2495 GOTO 2440
2500 IF D(5 <22 THEN GOTO 2560
2510 IF D(3 <N0 THEN GOTO 2530
2520 GOTO 2560
2530 LET D(5,=D(5)-10
2540 LET D(3)= D(3
2550 GOTO 2440
2560 LET E=N0
2570 IF D(5)=21 THEN GOTO 2730
2580 LET T=N1
2590 IF NOT D AND NOT E THEN GOT
O 2730
2600 IF NOT D THEN GOTO 2620
2610 GOTO 2640
2620 LET T=N0
2630 GOTO 2430
2640 IF USR HR THEN PRINT AT 84,
56, HIT?
2650 LET Z$ INKEY$
2660 IF Z$<>"Y" AND Z$<>"N' THEN
GOTO 2650
2670 LET D=Z$="Y"
2680 IF NOT D THEN GOTO 2400
2690 IF USR HR THEN PRINT AT 84,
56 'OK..
2700 GOSUB 200
2710 IF D(6)>21 THEN GOTO 2800
2720 GOTO 2410
2730 IF D(6)=21 THEN GOTO 3000
2740 IF D(6)<12 THEN GOTO 2760
2750 GOTO 2780
2760 LET D 6)=D 6)+10*D(4)
2770 IF D(6 11 THEN GOTO 2730
2780 IF D(5 >21 THEN GOTO 3000
2790 IF D(5 =D(6) THEN GOTO 3200
2795 GOTO D 5)>D(6))*100+3000
2800 LET Z$=' - BUST--- '
2810 GOSUB JAL '400'
2820 GOTO JAL 3100
2900 LET P=2*P
2910 LET Z$=W$
2920 LET D(6)=D(6)+10
2930 GOSUB VAL '400"
3000 LET Z$ 'WIN'
3010 LET O=O+P
3020 GOTO 3220
3100 LET Z$="LOSE
3110 LET O=O-P
3120 GOTO 3220
3200 LET Z$= PUSH
3210 GOTO 3220
3220 LET Z$='YOU +Z$+ +N$
3230 GOSUB 500
3240 GOSUB VAL 400
3245 GOSUB VAL '500'
3250 IF USR HR THEN PRINT AT 72,
60 D(5 AT 96 60 D(6 ,AT 84,N0
**$ O **
3260 IF O N1 THEN IF USR HR THEN
PRINT PRESS P TO PLAY
3270 LET T=N0
3280 FOR A=2 TO D(N1
3290 LET D N1)=N0
3300 LET M=A A N1
3310 LET I=A(A 2)
3320 IF INKEY$= P THEN GOTO 340
0
3330 GOSUB 10
3340 NEXT A
3400 IF O<N1 THEN GOTO 3450
3410 IF INKEY$<>'P' THEN GOTO 34
10
3420 GOTO 2200
3450 GOSUB VAL "500
3460 IF USR HR THEN PRINT "MONEY
ALL GONE. NEXT SUCKER?
3470 IF INKEY$="P" THEN GOTO VAL
"2010"
3480 IF INKEY$<>"O" THEN GOTO VA
L "3470"
3490 CLS
3495 IF USR HR THEN RETURN
3499 STOP
8900 SAVE BLACKJACK'
9000 REM
9001 IF USR HR THEN LPRINT J, '
, 00,00,00,00,00,00,00,00FF,61,
2E,10 13 0F,15,15FF,04,DB,20 8E
,FF,40 40FF,10,8E,81,39,FE,03,4
2,E0 C0,82,02 05,05 05,85 00 00
,00,00,00,00 03,04,15,15,25 29,2A
,EA,2A AA,5A,44 40,40 40,80,8
4,92 AC,92,42,42 22 A2,66,85,85
85,85 85,C5,A5,A5
9002 IF USR HR THEN LPRINT U =
,'05,04,02,01,00,01,0F,3D 2A,CA,
32,C4,39,C7,77,DD,E8,82,80,83,80,02
,98,26 52$0A,E1,B9 00,48 A3,54,4
9 A5 65,25,A5 45,35,BD 5D,77 DD,
F7,DD F6,DD FA,D5,77,DD 7A,D5,AA
,55,AA,55)2C,91,AE,40,AA,55,AA 5
5)A6,11,0E,40,AA 55,AA 55(B5,5D,
0D,55 AF,5F,AA,55
9003 IF USR HR THEN LPRINT U,
,'AA,55 FA F5,AA B0 BA AD+AA,55
AA,55 02 70 88,65-AA 55,AA 55 02
75 89 34+AA,55 AA 55 AB 5E BB,E
E,AB 5F BB,6F BB,EF BB EE BA BC,
AC,A2 A5,A4 A6 A5,92 2A C5 12,00
,9D 67,50,4A,A4,19 40,01 C1 01 4
1006,EE,E3,9C,23 4C,53 541BC F0
80,00,80 40,20 A0'
9004 IF USR HR THEN LPRINT U "2"
A5,A5,A3,A1,A1,A1 A1,A1368,45
44,42 42,49 35,49421 01,02,02 02
,22,5A,22555,54,57,54 94,A4,A8,A
8620,C0,00,00,00,00,00,007A1,A0,
A0,A0,40 41 03,07842,C0 7F,9C,81
,76 08,FF902,02,FF 71,04,DB,20 F
FAA8,A8,F0,C8,08,74,86,FF'
9006 IF USR HR THEN LPRINT J 8'
,"01,06,18,20,20,24 4A,C4CFE,21,
1C,08,04,92 A9,91D7A,E4,34,34,B2
,99 58,58E00,80 40,20 40 40 80,4
0F01,02,05,02 05,09 01 02G41,C1
41,C2 42,21 60,60H01 01 01,01 01
,81 01,01I4C,4C,4C 56 56,56 53,2
BJ20,10,08 64,9C 62 00,00
9007 IF USR HR THEN LPRINT J, K
,"02,04,05,02,01,86,0B,1EL93,B1,
48,38 D4,22 51,28MF2 C2 04 06 08
,11 E2,02NAB,A9 A9 A5 F4 82 4A M
5011,84 8A,84,15 04 04 02P38 6E,
BB EE,BB,EE,BB EE015,6A,85 C0 A1
,E2,B4,E6R55,AA,55,00 11,22,44 8
8555 8A,01 06,1B 2E,5B,86T02,80
C0,E0,B0,E8,BC,EE'
9008 IF USR HR THEN LPRINT U, L
, 77,3D,17,0D,07,03 01 40V61 DA
,74,D8,60 80,51,AAL11,22 44 88,00
,AA,55 AAX17,2D,47 85,03 A1,51,A
8Y77,DD,77,DD,77,DD,76,DCI40,20,
20,A8,21,51,21,88A5,52,4D,2F,A5
,95,95,D5,40,47,88,10 60 20,43 4
F,14,8A 44,2B,1C,12 8D,C9,78,D0
60,80,40,A0,20,40
9009 IF USR HR THEN LPRINT U,"
, 00,00,41 39,26,10 06,04 D4 CA
6A,6A,6A 32,32,32,80 80,81 80 80
,80,80 80,06,06,84 42 43 82 83 8
2,40,60,90 A0,40 A0 40 60,02 01
02,02,04 02,01,00,1A 1A 99 4D 2C
,2C,27,9E,89,95,09,20 D0 38 84 7
F,23 54,24,04 04,18 E0 00
9011 IF USR HR THEN LPRINT U
"FF,80,81 7E,40 7F 55 55FF 02
FC,04,FC,40,40,41FF,AA 04 F8 08
,08 C8 68,00,00 00,02 05,05,1D,2
5,00,00 00,01 19,25 2B,26,95 A5,
A9,2A 4A,52 54,54,40 40 40,40 81
,80 80,80C4,04,02 02 09 C7 28,0
8,05 05,05 05,05,05 05 05"
9012 IF USR HR THEN LPRINT ,
10 1F,0C 07,02 06 0B,1E,95,25
49,8A,F2 0D A0,E4,00,00 E0 1F 00
,00,80,41,38,64,04,02 FC,08 8C 0
6,A2,E9 A4,EA B9,E8 BB E8,22 14
88,49,2A,88,88,88,23 46 9B 2F 4E
,AE 9B,86,05 85,C5 E5 85 E9 BC E
E"
9013 IF USR HR THEN LPRINT ,
,"77,BD,97,AD,A7,A3 A1,A0,61,D9
,75,D2 74,D9,62,C4,11 11 11,54 92
,11 28 44,17,1D,17 9D,57,25 9, 4
5,A0 A0,A0 A0,A0 A0 A0 A0,60 31
10,3F,40,20,26,1C,82,01,00,00,F8
,07,00,00,27,05,B0,4F,51,02,A4,A
0,78,D0,60,40,E0,30,F8,08"
```

```
9014 IF USR HR THEN LPRINT U,"
 "10 14,E3,90,40,40 20,23 01 01,
01,81,02,02,02,02 2A,2A,4A,52,54
 95 A5,A9 64 D4,A4 98 80,00,00 0
0 A4 68 A0,A0,40,00 00 00,6 1
10,10,1F,80 55 FF 82 02 02 3F
3F 40,FF HA,AA,FE 02 7E,61 01 F
F
9016 IF USR HR THEN LPRINT
, 10, 3 7C 7C FE,D6 10 38 00
38,D6 FE,D6,10 38 00 44 EE BA D6
,6C,38,10 00 10 38 6C C6 6C 3
0 1C 03 88 7F 3E,CE 1C 08 1C 08
6B 7F,6B,1 1C 00 08 1C,36 6B,5D
,77 22 00 08 1C 36,63 36 1C,08,0
0 44 82,29 11 29 82 44 39
9017 IF USR HR THEN LPRINT
, 1F, D,70,66 6B 46 40,21 FC FE,
07 33 6B 31 91 42 21 10 12 09 04
,0A,31 40 42 04,A4,C8 10,25 C6 0
1 00,00,00,00,00,00 00,0F 49,88,
84 84 82,82 81 FF C9,6a 90,90 A0
,24,42 F9 09 09 9D A9 CF 90 20 4
F"
9020 REM
```

```
9021 IF USR HR THEN LPRINT 5D 1
"00,10,28,44,7C,44,44,00 00 38,4
4,04 38,40 7C 00 00 7C 04,15 04
44,38 00 00,05 18 28 48 7C,08,00
00 7C,40,78 04 44 38 00 00,38,4
0,78,44,44,38,00 00,7C,04,08 10,
20,20 00 00 35 44 38 44 44 38 00
00 38 44 44 3C 04 38 00 00 46 C
9 49,49 47,E6 00 00 1C,08 08 48
48,30 00 00 38 44 44 54 46 34 00
00 44 48 70 50,48 44 00
9022 IF USR HR THEN LPRINT 5D 14
'00 2C 22,2E 22 14 06 00 00
02 1C 20,22 1C 00 00 1C 22 10 1E
,20 3E 00 00 10 3E 12 14,18 10 0
0 00 1C 22,20 1E 02 3E 00 00 1C
22,22 1E 02 1C,00 20 04 04 08 10
,20 3E 00 00 1C 22 2E 15 22 1C 0
0 00 1C 20 3C 22,22 1C 00 00 67
92,92 92 93 6C 00 00 0C,12 12 10
,10,38,00 00 2C,12,2A 22 22 1C,0
0 00,22 12,0A 0E 12 22,00
9050 GOTO 2000
9100 REM
9105 FAST
9115 POKE VAL "32600 N1
9120 IF USR HR THEN SAVE BLACK
ACK" P
```

```
9130 IF PEEK VAL "32600 =N1 THEN
 IF USR HR THEN SAVE DATA
9140 IF PEEK VAL "32600 N1 THE
N IF USR HR THEN LOAD "DATA
9150 GOTO VAL "2000
 REM
9902 CLEAR
9905 LET HR=VAL "19400
9907 LET N0=NOT PI
9910 LET N1=NOT N0
9915 LET T=VAL 8192
9920 DIM C$(4 13 8,5
9925 LET E$=
9930 FOR X N1 TO 4
 FOR C=N1 TO 13
9940 FOR L=N1 TO 8
9945 FOR A=N1 TO 5
9950 LET C$(X C L A =CHR$ PEEK T
9955 LET T T+N1
9960 NEXT A
9965 NEXT L
9970 NEXT C
9975 NEXT X
9980 LET U$="
G000 REM
```

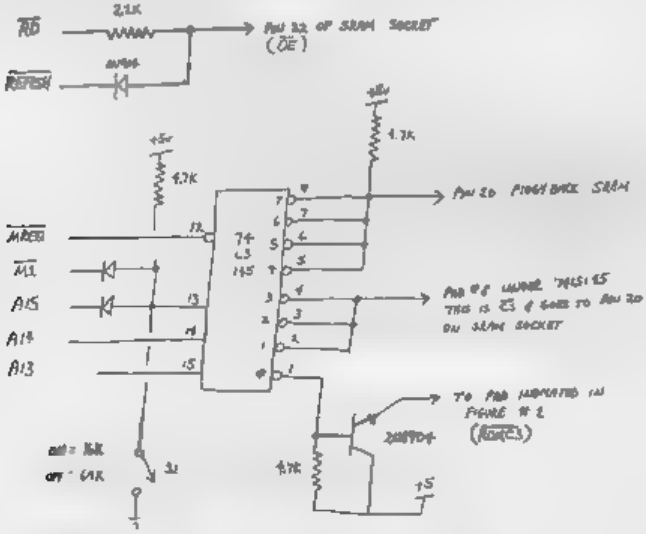
DELETE AFTER RUNNING

INTERNAL 64K RAM for the TS/ZX

by Tim Stoddard

I presented, near the end of 1986, two articles on upgrading the TS2016 ram pack to 64K. Shortly before that I had also designed a three chip INTERNAL 64K upgrade. But the chips used (NEC's 43256 SRAM) cost $55 EACH! This, I thought was much too much to pay for that amount of memory, hence, the article for the TS2016 upgrade. I have been following the price of the 43256 chip, and it is currently priced at $14.95, with every indication of going down further. So this is, perhaps, an opportune time for this article.

Schematic #1 shows the selection logic used. I used the 74LS145 which is an open-collector output and eliminates using diodes to tie together the apropriate 8K blocks for each of the two 43256 SRAMs. The 2.2K resistor and 1N914 diode at the top-left side of the schematic makes this upgrade compatable with VEYNIL's high resolution program THRUST. THRUST and this upgrade works BEAUTIFULLY! The upgrade must be in the 16K mode to run THRUST, however. The 2N3904 transistor deselects the BASIC ROM unless the current address is in the first 8K block.



Schematic #1

Figure #1

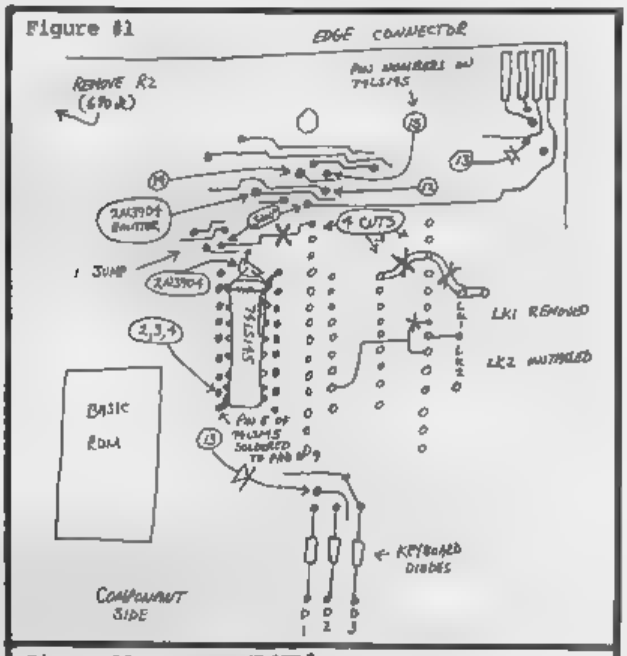
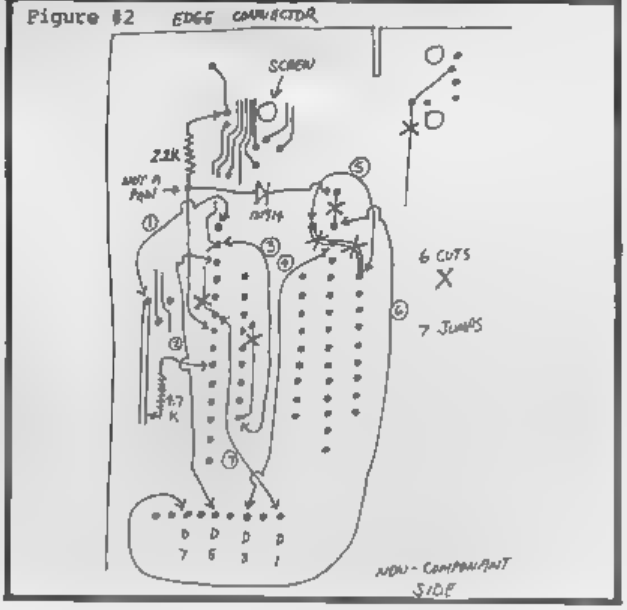


Figure #2



WARM UP THE SOLDERING IRONS'

First, using the proper anti-static procedures, take off the back of the TS/ZX case and remove the PCB from the case. The modification can be done with the keyboard connected but I don't recommend doing it that way. With the PCB out and on the anti-static mat, remove the old SRAMs and sockets if there are any. If you are modifing a 1K ZX81, remove jumper LK1, and install LK2.

Next, perform the four cuts shown in FIGURE 1, then install a 28 pin socket where the 2K ram would go. Using FIGURE 3, prepare the 74LS145 as shown and install it where the LEFT 1K SRAM would normally go. You'll need to slightly spread pin 16 up to pad 16 and solder, then spread pin 8 down to pad 9 and solder. This brings in the 5 volts for the 74LS145 chip.

Connect up the remaining wires, diodes, and transistor as shown in FIGURES 1 & 3 then reconnect the keyboard cables (carefully) if you have disconnected them. Turn the PCB over and perform the six cuts & 7 adds as shown in FIGURE 2. NOTE: Adds numbered 1 & 5 should be slightly larger wires than the standard 30 guage wire-wrap type wire, as these pass 5 volts on to other logic. Now add the two resistors and one diode as shown in FIGURE 2.

Plug in ONE 43256 SRAM into the 28 pin socket, screw the PCB into the top half of the case and power up. You should get the "K" cursor in a few seconds. Execute PRINT PEEK 16388 + 256 * PEEK 16389. This should give you 32768. If this works, power off, remove the 43256 SRAM and perform the "piggy-back" modification shown in FIGURE 4. Install the "piggy-back" SRAMs into the 28 pin socket and connect up the top SRAMs' CS wire to the 74LS145 as shown in FIGURE 3. Power up and execute one at a time.

```
1)POKE 16388,255
2)POKE 16389,255
3)NEW
4)PRINT PEEK 16388 + 256 * PEEK 16389
```

This should give you 65535 . . .INTERNALLY' Make sure that switch S1 is off (open). You can mount S1 anywhere you have room for in the case.

The SRAMs are available from MICROPROCESSORS UNLIMITED for $14.95 each plus $6 second day Federal Express delivery. They don't have a minimum order restriction and can be reached at (914) 267-4961. I would expect the price to drop another two to four dollars by the time this article is in print.

Tim Stoddard
85-48 66th Road
Rego Park, NY 11374
CompuServe 73127,2664

Figure #3

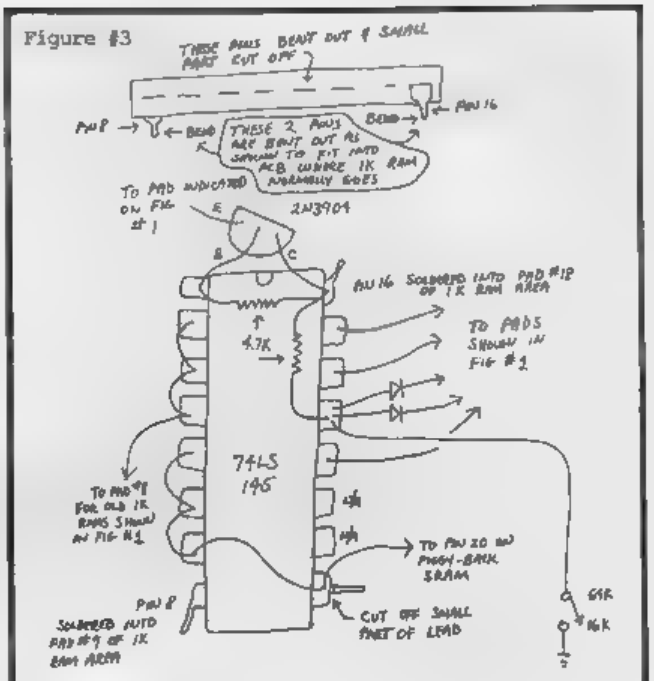
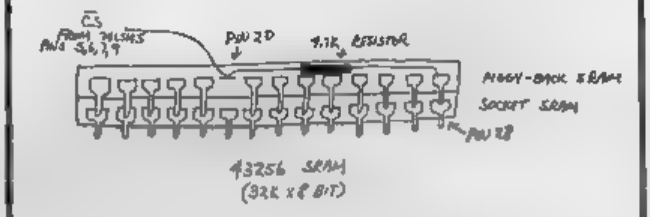


Figure #4



A STUDY IN NUMBERS

by Syd Wyncoop

Why would we want to study numbers? Because our computers only understand numbers. You may saying, "My computer knows Basic" or any of several other languages but, the truth of the matter is that it does not know or understand any language other than the assembly language of the Central Processing Unit (CPU). All languages and hence, all programs are stored in the computer memory as numbers. Look, for a moment, at your Sinclair character tables. The Basic command PRINT is really the number 245. Since all programs are numbers, it behooves us to know a little bit about them. I don t know about you, but, I hate machines that are smarter than I am!

All languages used by any computer, except assembly, are referred to as 'high level' languages. High level languages come in two varieties: compiled and interpreted. Compiled languages are converted (compiled) into the CPU's assembly language and run as machine code and therefore run quicker than interpreted languages. Interpreted languages are interpreted into assembly language at the time of execution. They still run as machine code but, much slower due to the interpreter processing time which uses a library of routines, which are slower by their very nature.

We do not need to learn assembly programming to use our computers however, understanding the numbers and why different bases are used will help us be more efficient programmers in the language of our choice. Of course, many of you may have desired to know more about machine code and been afraid to tackle it. After all, those long Hex (whatever that is) dumps in the magazines seldom make sense. Or maybe you have seen those Hexideoimal numbers and wondered why anyone would resort to such mutterings. If this discribes you, then perhaps the following study can be of help.

We are going to attempt to thoroughly understand three number systems or bases as they are properly called. There is a fourth base, Octal, which I will not discuss here as you will seldom encounter it. Octal will be found in the programming

guides published by chip manufactures and can be useful however, we can do quite well without it for this study Each base will be discussed seperately however, I will make referances back to Decimal, as that is the one we all use dailey

Decimal (Base 10)

The first is Decimal (dec), or base 10 numbers. We are all familiar with decimal as we use it every day to count our bags of money and for various other tasks Most of us learned in school that the columns of digits represent ones, tens, hundreds, etc For example

```
t
h b
o u
 u n
  s d
   a r t o
    n e a n
     d d n s
      s s s u

      1 2 3 4
```

What you may not recall is, each column represents 10 (the base)raised to the power of the number of the column as counted from right to left, starting at zero Hows that for doubletalk?

Follow through this example, to see that 1234 decimal really means:

```
10^0 =    1*4 =    4
10^1 =   10*3 =   30
10^2 =  100*2 =  200
10^3 = 1000*1 = 1000
                -----
Total=          1234
```

Study this carefully as it is the easiest example we will have and it must be understood or the rest will really seem like Greek to you.

Before continuing, it bears mentioning that all bases are represented by the digits 0 to Base-1 Base 10 is therefore represented by the digits 0 to 9 It is these digits which then represent a mutiple of a power of the base, as above (10^0=4). No digit can be greater than Base-1 because at the point it equals the Base there is a carry to the next column For instance, in base 10, 9+1=0 and carry 1 to next column. All of this should be familiar to you but if you are like me, you haven't given it much thought since school No,I wont say how long ago that was!

Binary (Base 2)

Now for the hard stuff, Binary (bin), or base 2 Following our discussion of the last paragraph, we can only represent binary numbers with the digits 0 and 1 (remember base-1) This means our columns also must have meanings other than ones tens etc They now become ones, twos, fours, eights, sixteens etc which are the powers of 2 (our base) instead of powers of 10 For example

```
o
 l f
  g o t o
   h u w n
    t r o e
     s s s s

     1 0 1 1
```

As in our previous example, 1011 really means

```
2^0 = 1*1 = 1
2^1 = 2*1 = 2
2^2 = 4*0 = 0
2^3 = 8*1 = 8
            -
Total=     11 decimal
```

Now you know why we count in decimal! You thought it was because we have ten fingers Imagine having hands with two fingers on each hand Binary would then seem as easy as decimal If you have any difficulty with this, go back to the discussion on decimal and compare it with this one. Only the base is different

What you have just learned is how to convert binary to decimal The procedure for converting decimal to binary is similar. Briefly, divide your number by the largest power of two not larger than your number You continue this process with successively smaller powers of two until you have reached 2^0, at which time there should not be a remainder Write down (left to right) a 1 when the division is possible and a 0 when not possible. Using our example of 11

Step	Do	Result
1	2^4=16 and 2^3=8 therefore, 2^3 or 8 is the number we want.	
2.	2^3=8 and 11/8=1 remainder 3	1
3	2^2=4 and 3/4=0 remainder 3	10

```
4   2^1=2 and  3/2=1 remainder 1          1Ø1
5.  2^Ø=1 and  1/1=1 remainder Ø          1Ø11
6   We have now converted 11 decimal to 1Ø11 binary
```

Some of you may be wondering what the point of all this is After all I barely passed math in school. why bother with this now? The point is, while decimal is more comfortable for us humans, binary better represents how our computer 'thinks'.

An explanation of the CPU is in order This is background only to give you some understanding why the 'smart' computer doesn't understand decimal. The CPU (which is the Z8Ø in our Sinclairs) is merely a collection of transistors and transistors are simply electronic switches. For those of you who know better, please bear with me, my end will justify the extreme over simplification. We all know that a switch can either be on or off Binary allows us to represent the on/off states with one and zero, respectively. Not quite perfect but it works

Most binary numbers you will see, will have eight digits or some multiple thereof. This is accomplished by padding out the number with leading zeros. For instance, 1Ø11 binary would normally be written as ØØØØ1Ø11b The reason for eight Binary digITS (bits) is that eight bits make one byte

A byte is not what your neighbors' dog puts on you A byte is the 'wordsize' of your CPU. A word is the number of bits handled as a complete unit by the CPU and is commonly referred to as a byte The Z8Ø is an eight bit CPU therefore one byte=eight bits Words and bytes are not exactly the same, but will suffice here You do not need to understand the internal workings of the CPU in order to understand binary numbers

Binary is most useful for graphics in Basic or masks in assembly Unless you decide to learn assembly you will seldom work directly in binary due to the difficulty keeping the digits correct, which leads us to the next base

Hexadecimal (Base 16)

Now that we have mastered binary we can dig into the last of the three number systems Hexidecimal (hex) or base 16 is used because it works very well as a shorthand for that awful binary I have provided several charts for easy conversion from hex to decimal and hex to binary or vice versa So don't leave now, things are about to make sense

One challenge we have with hex is that there are not enough digits for Ø to base 1 We have only Ø to 9 which will work fine for the first ten digits We need to improvise for the last six digits and someone far wiser than me (us?) has already solved our dilemma. The digits needed to represent 1Ø to 15 are A to F The sixteen hex digits are now Ø, 1, 2, 3, 4, 5 6, 7, 8, 9, A, B, C, D, E & F.

Now that we have the digits, we need to recognize the form taken by hex numbers. Just as binary are usually eight digits long, hex numbers are always two (or some multiple of two) digits long This makes counting in hex as follows:

```
ØØ,Ø1,Ø2,Ø3,..  ,ØA,ØB,ØC,ØD ØE,ØF,
1Ø,11,12,..  .          ,1D,1E,1F,
2Ø,21,..     .          .  ..2F,
3Ø,..  .  .             .
.. ..        .          ..  FF
```

A close look at the hex to decimal conversion chart will make this much clearer

You may occasionally see hex numbers that have an odd number of digits The first digit will be a zero and the second a letter, such as ØFFh. I do not use a leading zero however, be aware some assemblers require it, therefore you may see it

There is no need to go into the math needed to convert between hex and decimal as the chart provided will serve the purpose much better, easier and faster Those of you interested in working out the details need only follow the examples for decimal and binary. Keep in mind the base is now 16 therefore,the columns of digits will represent ones sixteens, two hundredfifty-sixes, etc

All this brings us to what I mean by hex being a shorthand for binary and the reason we are even interested in hex. A close look at the hex to binary conversion chart will make this more obvious You can readily see that four bits can be represented by one hex digit For those of you who understood my ramblings about bytes, four bits is a small byte (also known in some circles as a nybble) Could I make this up? Therefore, we can represent any eight bit byte with only two hex digits.

You may recognize that this is not that much better than decimal however, decimal cannot be converted to binary with the same ease as hex Also, numbers larger than 255 will really create some headaches that hex helps solve (more on this in a minute).

We now know how to write numbers in three bases, decimal,binary, and hex. In order to avoid confusion we need to make a proper designation of each You should always suffix all binary numbers with a 'b' (1ØØØ1ØØb) and all hex numbers with a 'h' (25h) Occasionally, you may see a 'd' suffix on decimal numbers however it is not needed as decimal is the default We will do this even though some numbers can obviously only be hex (FFh) You must always be careful to follow this notation or you will create unnecessary confusion for all.

You should take some time to practice using all the charts Also, practice simple arithmetic in each base (add, sub, mult & div) You can use the charts to check your answers. Before long you will be thinking in hex and binary as easily as you now do in decimal

Ok, let's look at how numbers larger than 255 are stored and handled by the CPU.

That's twice I mentioned 255 without an explanation. The reason 255 is a magic number is because it is base-1 for base 256 numbers. I am not going to boggle your mind with this number system as it is not needed by us, only the CPU uses it. Why in the world does the silly CPU use base 256? Let's go back to our discussion of binary, bytes and related whatevers. Remember, we padded out our binary numbers to eight digits as one byte for the Z8Ø is eight bits? The reason the CPU uses base 256 is 2^8 = 256 Voila! On your own, figure out what 11111111b means in decimal

We can actually store numbers between Ø and 65535 by tieing two bytes together This is done by the CPU automatically to generate addresses The second byte is increased by a factor of 256 as this is the number that generates a carry out of the first byte. For example.

```
  1 1 1 1 1 1 1 1
+             1
-----------  .
1 Ø Ø Ø Ø Ø Ø Ø Ø
```

Remember, 1+1 binary = Ø and carry 1 just as Ø+1 does in decimal Work out the above problem yourself to see how the result is achieved You will notice we now have a number which is nine digits long In truth, the actual number is ØØØØØØØ1ØØØØØØØb because we are tieing two bytes together and each byte is padded out to eight digits Since our number is now sixteen bits long, the largest number we can store is 2^16-1 or 65535.

You can now see why we need a shorthand for binary Today, we rarely work with large binary numbers as it is too easy to err We will not discuss binary numbers larger than eight digits either. If you decide to learn an assembly language, you will probably only use binary when working with the logic instructions. You may find it interesting that early programmers had to use only binary numbers and they were entered from a panel of switches instead of a keyboard. Today, using machine code is duck soup and hex is much easier to work with.

What did I mean, 'tie two bytes together and increase one of them by a factor of 256?' Let's assume our number is stored in byte 1 and byte 2. The formula to recover our number is

Peak byte 1 + Peak byte 2 * 256

Looks a little familiar? You probably have seen something similiar before and did not know what was happening The byte we increase by the factor of 256 is called the high byte which makes the other byte the low byte. Using high/low nomeclature our formula becomes

Peak low byte + Peak high byte * 256

One peculiarity designed into the CPU is that contrary to the number systems we have discussed, the CPU stores the low byte first. This must be kept in mind or you will not at all get the results you were trying to achieve. Scan the list of system variables in your Sinclair manual and use this formula on some of the 2 byte variables The results are the address at which that area of memory begins

Review some of the areas of your Sinclair manual that did not make sense before. Especially the chapters on number systems, machine code, system variables, memory, and the appendices. There is a wealth of information there however, it is presented so poorly that it may not have made sense before Then compare notes with this study and you will be well on the way to understanding machine code.

Hex/Bin Conversions

Hex	Bin	Hex	Bin
Ø	ØØØØ	8	1ØØØ
1	ØØØ1	9	1ØØ1
2	ØØ1Ø	A	1Ø1Ø
3	ØØ11	B	1Ø11
4	Ø1ØØ	C	11ØØ
5	Ø1Ø1	D	1111
6	Ø11Ø	E	1110
7	Ø111	F	1111

To use any four bits can be represented by a single hex digit. For example, 1Ø1ØØ1ØØb = A4h

Hex/Dec Conversions

	Ø	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
Ø	Ø	1	2	3	4	5	6	7	8	9	1Ø	11	12	13	14	15
1	16	17	18	19	2Ø	21	22	23	24	25	26	27	28	29	3Ø	31
2	32	33	34	35	36	37	38	39	4Ø	41	42	43	44	45	46	47
3	48	49	5Ø	51	52	53	54	55	56	57	58	59	6Ø	61	62	63
4	64	65	66	67	68	69	7Ø	71	72	73	74	75	76	77	78	79
5	8Ø	81	82	83	84	85	86	87	88	89	9Ø	91	92	93	94	95
6	96	97	98	99	1ØØ	1Ø1	1Ø2	1Ø3	1Ø4	1Ø5	1Ø6	1Ø7	1Ø8	1Ø9	11Ø	111
7	112	113	114	115	116	117	118	119	12Ø	121	122	123	124	125	126	127
8	128	129	13Ø	131	132	133	134	135	136	137	138	139	14Ø	141	142	143
9	144	145	146	147	148	149	15Ø	151	152	153	154	155	156	157	158	159
A	16Ø	161	162	163	164	165	166	167	168	169	17Ø	171	172	173	174	175
B	176	177	178	179	18Ø	181	182	183	184	185	186	187	188	189	19Ø	191
C	192	193	194	195	196	197	198	199	2ØØ	2Ø1	2Ø2	2Ø3	2Ø4	2Ø5	2Ø6	2Ø7
D	2Ø8	2Ø9	21Ø	211	212	213	214	215	216	217	218	219	22Ø	221	222	223
E	224	225	226	227	228	229	23Ø	231	232	233	234	235	236	237	238	239
F	24Ø	241	242	243	244	245	246	247	248	249	25Ø	251	252	253	254	255

To use: find your decimal number, follow to left edge for first hex digit and up for second hex digit. For example. 237 = EDh.

NUMBER BASE CONVERTER

by Syd Wyncoop

This program runs on the TS2068 or the ZX81

This is a useful BASIC routine to do conversions between number bases. It will run on both the ZX81 and TS2068. I have included in the listing the necessary changes for the ZX81 (or TS1000/TS1500). The only other requirement to run on the ZX81 would be to break the multi-statement lines into all single statement lines.

```
  1 BORDER 0  PAPER 0  INK 4  C
LS
  2 ON ERR  GO TO 800
  3 GO TO 1000
 100 REM Decimal to Hex
     Entry a=decimal number
     Exit  a=decimal number
           h$=hex number
 110 LET h$="": LET hh=INT (a/25
6): LET hl=a-hh*256
 120 IF hh<1 THEN  LET h$="00"
GO TO 140
 130 LET n=hh: GO SUB 150
 140 LET n=hl
 150 LET n1=INT (n/16): LET n2=n
-n1*16
 160 LET h$=h$+(CHR$ (n1+48+(7 A
ND n1>9)))+(CHR$ (n2+48+(7 AND
n2>9)))
 170 RETURN
 180 REM   TS1000 users replace
line 160 with. LET h$=h$+CHR$ (
n1+28)+CHR$ (n2+28)
 200 REM Hex to Decimal
     Entry:a$=hex number
     Exit  a$=hex number
           a=decimal number
 210 LET h$=a$: LET a=0
 220 IF LEN h$<3 THEN  GO TO 250

 230 GO SUB 250
 240 LET a=a*256: LET h$=h$(3 TO
 )
 250 LET a=a+(CODE h$-48-(7 AND
h$>"8"))*16+(CODE h$(2)-48-(7 A
ND h$(2)>"9"))
 260 RETURN
 280 REM   TS1000 users replace
line 250 with. LET a=a+(CODE h$
28)*16+(CODE h$(2) 28)
 300 REM Decimal to Binary
     Entry a=decimal number
     Exit  a=decimal number
           b$=binary number
 310 LET b$="00000000"  LET n=a
 IF n>255 THEN  LET n=INT (n/25
6)  LET n1=a-n*256  GO SUB 360
 LET a$=b$  LET B$="00000000"
LET n=n1  GO SUB 360  LET b$=a$
+b$  RETURN
 380 FOR i=7 TO 0 STEP -1
 370 IF n>=2^i THEN  LET b$(8-i)
="1"  LET n=n-2^i
 380 NEXT i  RETURN
 400 REM Binary to Decimal
     Entry a$=binary number
     Exit  a$=binary number
           a=decimal number
 410 LET b$=a$  LET a=0  LET n=L
EN b$ 1
 420 FOR i=n TO 0 STEP -1
 430 IF b$(i)= 1" THEN  LET a=a+
2^i
 440 LET b$=b$(2 TO )  NEXT i
 450 RETURN
 500 REM Set Decimal Tabs
 510 LET t=4
 520 LET T=T-(1 AND A>10000)-(1
AND A>1000)-(1 AND A>100)-(1 AN
D A>10)
 530 RETURN
 800 REM Error Trap
 810 PRINT #0;"Invalid Entry--Pl
ease Try Again"  PAUSE 150
 820 GO TO 1030
1000 REM Main Loop
1010 PRINT ''TAB 8,"NUMBER CONVE
RTER"'''
1020 PRINT "Be sure to append a
'B' or 'H'"'"to all binary and
hexidecimal"'"entries, respecti
vely."'"There is no suffix req
uired for decimal numbers, as t
hey are thedefault "
1030 POKE 23658,8: INPUT AT 0,0,
"Type number and press ENTER"'
LINE a$  LET n=LEN a$
1040 IF a$(n)="D  THEN  LET a$=a
$( TO n-1): GO TO 1070
1050 IF a$(n)="H" THEN  LET a$=a
$( TO n-1)  GO SUB 200. GO TO 1
060 IF a$(n)="B  THEN  LET a$=a
$( TO n-1). GO SUB 400  GO TO 1
080
1070 LET A=VAL A$
1080 GO SUB 100  GO SUB 300  GO
SUB 500
1090 PRINT AT 16,0,  Dec      Hex
     Bin '"----------------------
------------
                "
1100 PRINT AT 18,T,a;TAB 8,h$;"h
",TAB 15,b$;"b"
1110 GO TO 1030
9990 MOVE "NumConv bas",1
```

Beginning Z80 Machine Code

LESSON EIGHT

By Syd Wyncoop

Last time we covered a lot of ground by skimming quickly over it. If you did not understand all the rotate and shift instructions, don't worry. You can get along fine without them for now. You should however make sure you hover over the logical instructions, And, Or and Xor, long enough, as they are very useful.

You will also need to review my article on number systems, as a thorough explanation of Binary is given there. For this reason, I did not give much explanation of Binary in the last lesson.

Now, lets discuss some instructions that are off in the corner by themselves. The first is Nop, which is read No OPeration. You guessed it, it doesn't do anything. Almost. The purpose of Nop is to waste time (4 T states) and provide a space filler. No flags are affected.

Nop is also used by us programmers to fix our errors. You can replace unwanted instructions with a Nop, by direct poking, rather than re assemble a large section of code.

Cpl means ComPLement the accumulator. We have briefly mentioned complementing a few times. The result is to invert all bits. For example, assume the accumulator contains 10111011b. After complementing, the accumulator will contain 01000100b.

Neg will NEGate (2's complement) the accumulator. Negating the same value of 10111011b in the accumulator will yield 01000101b. Creating the two's complement of a number is done by complementing the number and adding one. Neg performs this operation on the contents of the accumulator. Any number plus its two's complement will equal zero.

The last instruction is Daa which means Decimal Adjust Accumulator. This instruction works in conjuction with the Rld and Rrd instructions from last lesson. As I stated then, I have not found a good use for these instructions. I will however give a brief description of their purpose, as you may encounter them.

Daa adjusts the accumulator for Binary Coded Decimal (BCD) addition and subtraction operations. In order to understand Daa and its rotates, we need to understand BCD.

BCD is a method of representing a decimal digit (0-9) in four bits, much the same as any single hex digit (0-F) can be. This means that 15BCD is 00010101b in binary digits and not the usual 00001111b that we would normally find in binary representations. Look again, this is because there can be no digit greater than 9 in any four bits, in BCD.

The problem is that our old friend, CPU, does not know whether we want to store decimal or hex digits in its words. Therefore, Daa will check the half-carry flag to determine when to adjust the binary result, back to a decimal digit. The only advantage to BCD, that I have found, is that it makes the retrieval of decimal digits easier.

As I have said, I do not subscribe to the philosophy that BCD is a necessary or useful tool. Therefore, I will dwell on it no longer. You can find texts on the subject, if you are interested.

That's all the instructions for this lesson. (Tim says I have been getting long-winded). I did however leave some of you hanging last lesson with the question, "What is source code?" This question was raised by some readers requesting information and instruction on the use of assemblers.

I must make very general comments as each assembler has its own set of commands. Unfortunately, the best way to learn the commands is to use the assembler. The basic operation of assemblers is simple. They allow us humans to type in our MC as mnemonics instead of hex digits. The assembler then uses our mnemonics to assemble the binary instructions that are understood by the CPU.

There are basically two types of assemblers. The first is an interactive or virtual assembler. a good example is Hot-Z by Ray Kingsley. This type of assembler is a little more difficult to use but much quicker. It generates the MC in place as you type in the mnemonics. The disadvantage is that it is not as easy to move and re-address large sections of code.

The second type will generate an ASCII source code file in addition to the assembled MC Source code is much like a letter you might type on your word processor. It is an ASCII encoded file containing all the labels and mnemonics you type into it, as well as pseudo-ops. The disadvantage is a loss of speed and disc space, as there are two files to handle Much of this is offset by the gain in flexibility to change the code as well as the ability to comment the source code file The addition of comments are a huge benefit in helping yourself remember what it was you had intended the code to do. The best example is Zeus by Crystal Computing

I use Hot-Z and Zeus regularly and feel they are the best available for the 2068 Hot-Z is the only assembler available for the TS1000 I have several for the TS1000 that put the mnemonics in REM statements, but none of they work very well

A moment ago I mentioned pseudo-ops. These will vary as to availability with your assembler and will be detailed in the instruction manual Pseudo-ops are assembler directives, not mnemonics They tell the assembler important things at the time of assembly They do not generate any MC instructions and are not executable by the Z80

Typical pseudo-ops are

```
        Org     assemble the following MC from this address
                to End or the next Org
        Equ     set label equal to specified value
        End     end of source file
        Defb    defines the contents of the current byte
        Defw    defines the contents of the current word
        Defs    defines an amount of bytes to be reserved
at
                the current address
        Defm    defines the contents of the current bytes
as
                the stated ASCII string
```

There are many more pseudo-ops, but these are the most common You will also encounter labels and comments in source listings Labels are much like Basic variables They assign a name to an address Some assemblers require labels to be followed by a colon

Comments are just that and are preceded by a semi-colon. They may appear at the end of a line, to explain that instruction, or on a line by themselves

Now, our utility for this lesson. Those of you using the TS1000 will particularly appreciate this one. It will generate a line 0 Rem statement of the size specified You need only type two commands to run the code

```
Rand length
Let x=Usr address
```

The first will load the length of the desired Rem statement into the system variable, Seed. No formatting or conversions required by you. The second statement will run the routine from the loaded address

Watch the source carefully, as it will run on both the the TS1000 and TS2068, but there are some lines that are unique to each computer. Also, it must be loaded and run from above ramtop or you will confuse the poor CPU

```
.Generate Rem Statement
.
GenRem  Ld HL,(Seed)        ,get length desired
        Inc HL              ,adjust for Rem
        Inc HL              ,adjust for EOL
        Push HL             ,save line length
        Ld BC 4             ,add overhead bytes
        Add HL,BC
        Push HL              get length into BC as required
        Pop BC              ,by Rom routine
        Ld HL,407Dh         ,TS1000 only--address to put Rem
        Ld HL (Prog)        ,TS2068 only--statement at
        Call 099Eh          ,TS1000 only  Rom routine to make
        Call 12BBh          ,TS2068 only--BC spaces at HL  The
                             routine returns with DE=last byte
                             of the new space & HL=first byte-1
                            ,of the new space
        Ld A,76h            ,TS1000 only--EOL marker
        Ld A,0Dh            ,TS2068 only
        Ld (DE) A           ,load end of line marker into the
                             last byte of the new line
        Xor A               ,A=0
        Inc HL              ,correct the pointer
        Ld (HL) A           ,load high byte of line number
        Inc HL               adjust pointer
        Ld (HL) A           ,load low byte of line number
        Pop BC              ,retrieve line length
        Inc HL              ,adjust pointer
        Ld  HL),C            load low byte of line length
        Inc HL               adjust pointer
        Ld (HL),B           ,load high byte of line length
        Inc BL               adjust pointer
        Ld (HL),EAh          load Rem character
                             the line is now complete  but it
                            ,contains  garbage  that should be
                            ,cleared out
        Ld BC,(Seed)        ,get length of space made
Loop    Xor A               ,A=0, you could use Ld A,x. where
                            ,x=any fill character
        Inc BL               adjust pointer
        Ld (HL),A           ,clear the byte
        Dec BC              ;adjust counter
        Ld A,B              ;and test if BC=0
        Or C
        Jr NZ,Loop          ;loop to clear next byte if BC<>0
Done    Ret                 ;return to Basic
```

Early in this series, I promised to show you an easy way to save and load your MC above ramtop on the TS1000 Since we now have a routine that must run there, this seems like a good time to keep my promise

Make the following changes to the hex loader from lesson 1

```
60 Let A$="218C40C95"
70 Rem insert your own MC string of hex digits
   between the quotes in line 60 and end the
   string with an 'e'
```

The secret to remember is that A$ contains all the MC as a string of hex digits You must also remember to reset ramtop and type a New command, before re-loading your saved MC Basic Next lesson we will learn the block instructions You will be amazed at the difference in speed

And now, a special little utility for the TS1000. The TS2068 has several nice graphics commands This routine will emulate the Flash command on the TS1000 by flashing the entire screen I will leave the modification of it to you, for flashing one or more characters

```
.Flash Screen
.
Flash   Ld HL,(D-File)      get start of D-File
        Ld C,18h            ,line count
OutrLp  Inc HL              ,step over EOL marker
        Ld B,20h            ;character count
InnrLp  Ld A,(HL)           ;get character
        Add A,80h           ,invert it
        Ld (HL),A           ,replace character
        Inc HL              ,advance character pointer
        Djnz, InnrLp        ,check if line complete and loop
                            ,if not
        Dec C               ,count line done
        Jr NZ, OutrLp       ,check if screen complete and
                             loop if not
Done    ■■■                 ,return to Basic
```

This routine can be called from your Basic programs with a Rand Usr address command The following will demonstrate

```
10 Rand Usr Flash
20 Pause 50
30 Goto 10
```

I hope you find these routines useful We have two more lessons to go and we will be done with the Z80 instruction set You should now be able to write your own MC routines I believe the only way you can learn MC is to use it, so get stumbling!

"THREAD SPOOLING" WITH THE CUMANA QL DISK DRIVE INTERFACE

by Joe Newman

They call it the "bargain" interface, or the "economy" interface. I prefer not to refer to it as this, because of the negative conotations of the words "bargain" or "economy". Sure the Cumana is the cheapest disk drive interface (along with the CST) for the QL, but it is more than just a disk interface.

Before I got my first Cumana interface, I didn't realize the power potential for this little black device. Only when I started looking through the nicely printed manual at the EXTRA COMMANDS that the interface adds to SuperBASIC, did I start to realize the Cumana's potential. I started to examine this potential, and discovered a few very interesting things.

In this article, we'll examine a useful and easy to use feature of the Cumana know as "Print Spooling". Print Spooling allows you to send a file to a "spool", which will then send all files on the spool to the printer. The printing is a background job on the QL...in other words, files print out while you still have complete control of SuperBASIC. This can save much time, especially if you have some long files to print out. You can send a file to the spool, and the file will be printing out while you are typing in some program lines! You are no longer limited to the buffer of your printer!

The example program I have made is for printing out more than one QUILL (QL Wordprocessor) file. Imagine that you have five or six letters to type, but don't wish to wait for each letter to print out while you type the next one. To use my program, you just PRINT your letters to disk, then use my program to print out all the letters at once, at your convenience.

In order to print a file to disk, first type your article, letter, or whatever, then choose the print command in Quill (Hit F3, then P). Now hit ENTER twice only! The prompts that you press ENTER to are: Current, and Whole. Now instead of hitting ENTER for printer, type in a name (such as "letter1"), then hit ENTER. You file will now be stored on disk in the exact format as if it were printed out. In order to be able to use my print spooling program, you MUST print all files you wish to spool to disk in that manner.

Once you have completed your typing and are ready to print your letters, load my program into your QL, RUN it and the following will happen: you will be asked "Number of files to print?". Enter the total number of files you wish to print at that time. You will then be asked "Drive to use (1 or 2)". Enter the number of the drive your disk with the files is in. You can enter a number greater than 2 if you have more than 2 drives. Next you are warned that any file which is spooled CANNOT be removed from the spool until it prints. There is no way to stop the printing of a file unless you reset the QL. So make sure you print the right files. Now you will be asked to enter the name of the file you wish to spool. Just enter the filename itself; the drive name and extension _LIS will be automatically added (the _LIS extension is added by Quill to all files which are printed to disk). Your disk should "whir" and the printer will start printing. If your printer was off, you can still turn it on now and you will not loose any data.

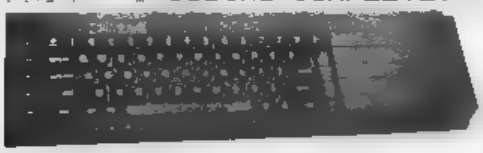
Let's now examine the program in more detail. Notice in Line 50 the FLP_EXT? This tells your Cumana interface that you wish to enable the extra commands. If this is not entered first, entering any of the extra commands will just return a bad line error report. Line 70 had the statement: DATA_USE DR$. This tells the Cumana which drive will be accessed by the spool command to find your files. DR$ will just be equal to FLP1_ or FLP2_ (or whatever drive you entered). Line 110 is the spool control...SPL N$. Yep, that's all there is to it! The N$ is just the name of your file (with the _LIS extension added). To spool any file, just type SPL and the name of the file. The DATA_USE statement specifies which drive to access for the file. After you have entered all your file names, the printer will be printing, and the flashing red cursor will be on the screen. Yes, you can now use your QL for other purposes, even while it is printing! Try loading a new program while something is being printed...it will work fine. The only real limitation to how much can be spooled or done while spooling is being carried out, is your QL's memory. If you have several large files being printed, you may not be able to load a new program, although you will still have control.

To check on the status of the spooled files, you can enter JOBS and a list of all your spooled files will appear on the screen in the order in which they were spooled. Suppose you send three files to be spooled: file1, file2, and file3. If file1 is currently being printed, and you decide you don't want to print file2 and/or file3, you can remove them by typing RJOB and the number under the "Job Number" column, a comma, the number under the "Job Tag" column (after entering JOBS), another comma, and a 0. This will remove that file from the spool. The following is an example of what is displayed when JOBS is used:

JOB	TAG	OWNER	PRIORITY	
0	0	0	32	
1	2	10	1	SPL
2	3	8	1	SPL
3	4	7	1	SPL

continued...

```
20 REMark        © 10-31 1986
30 REMark      BY JOSEPH NEWMAN
40 CLS
50 FLP_EXT
60 INPUT "Number of files to print?";x
70 INPUT "DRIVE TO USE (1 OR 2)";DRIVE:DR$="FLP"&DRIVE&" ":DATA_USE DR$:CLS
80 FOR loop=1 TO x
90 STRIP 0:INK 2:AT 0,10:PRINT "WARNING!":STRIP 2:INK 7:PRINT "Once a file is st
nt to the spool it CANNOTbe removed or stopped "
100 INPUT "PLEASE ENTER FILE NAME....(no ext or drivenecessary)";n$:n$=n$&"_LIS'
110 spl n$
120 CLS
130 END FOR loop
140 PRINT "SPOOLING COMPLETE."
```



I strongly suggest you do the following: Load the INSTALL_BAS program for Quill. Choose EDIT to edit your current printer driver, then change or add the POSTAMBLE CODE to the code for a form feed on your printer. Then INSTALL this printer driver again. The form feed will now be sent after every file is printed to make sure printing for each file starts on a new page.

If you have any questions or problems you can contact me at: 325 West Jersey Street #2D, Elizabeth, NJ 07202.

QL ABACUS/SPREADSHEET "TIPS"

PART I

by

Mike de Sosa

QL Abacus (QL Spreadsheet or QL-SS in its stateside version) is Psion Ltd.'s excellent spreadsheet program "bundled" with the QL. In the unlikely event you're unsure of just what such a program can do for you, a "spreadsheet" program facilitates the rapid entry of text, data, and formulas into the program and then provides automatically calculated results. Spreadsheet programs are frequently called upon to answer the question "What if?" regarding long and complex calculations: in this case, any one or more input values may be changed and the new result instantly recalculated. Spread sheet programs are the second most widely used computer program after word processor programs and one big reason for the popularity of personal computers today.

Spreadsheets are not new: the accountant's worksheet, the navigator's log, the builder's list of materials, and calendars and schedules of all types are forerunnners of the concept.

QL Abacus is one of the best spreadsheet programs--certainly one of the easiest to learn and use--and has few limitations. Some other spreadsheet programs have more rows and columns (QL Abacus provides 255 rows and 64 columns--16,320 cells); some have more directly usable commands and functions (QL Abacus could do with more preset statistical functions), and some are more fully integrated with database and other programs, but QL Abacus has enough of everything required for <u>almost</u> <u>any</u> appli cation.

GETTING STARTED WITH QL ABACUS

You can plan a spreadsheet on your computer screen, on plain paper, or on some type of graph or tabular paper. When you plan it, give thought to the export formats used to transfer data to the other "bundled" programs, particularly QL Easel (Business Graphics). Get your data and formulas onto the spreadsheet and verify your results without worrying too much about appearance (tabulation, justification, column width, labels, etc.). All of the latter may be taken care of later, if needed. You will usually be looking for a bottom-line result, and you only get points for neatness if the result is to be printed or otherwise displayed.

I won't go into the description of the QL Abacus screen, see the <u>QL User Guide</u> my new book available through <u>Time Designs</u>, or run it up on your monitor--it is similar in appearance to other QL software programs.

Enter data or a formula at the cursor location by typing it on the input line and keying ENTER (the program will automatically distinguish between data and formula); enter text by first typing a single or double quotation mark (it is not necessary to end text with a quotation mark). Data entered will appear at the cursor position, formulas will not--only their result. Text will be continued beyond the cell boundary, if necessary.

A stumbling block for many Abacus users is the concept of <u>relative</u> (the norm) and <u>absolute</u> cell references, the cell reference being the grid coordinates of the cell. Abacus assumes a relative reference, that is, if you enter the formula A1*A2 in cell A3 and then ECHO this formula in cells B3, C3, and D3, the formulas in each cell will be as follows.

CELL	A3	B3	C3	D3
FORMULA	A1*A2	B1*B2	C1*C2	D1*D2

Prefix cell references with a '$' symbol if they are intended to be absolute, for example,

CELL	A3	B3	C3	D3
FORMULA	$A1*$A2	$A1*$A2	$A1*$A2	$A1*$A2

Cell references may be mixed, for example,

CELL	A3	B3	C3	D3
FORMULA	$A1*A2	$A1*B2	$A1*C2	$A1*D2

and the '$' can be used with labels used as cell references, for example, $May.sales.

Another stumbling block for spreadsheet novices is the use of formulas. Formulas are distinguished from data in that they always begin with either a cell reference or an Abacus function. A1+A2 entered in cell A3 is a formula, as is INT(1+A2). Formulas may be difficult to understand and even more difficult to invent; following are some tips:

1) Break down complex formulas into more readily understandable segments and then link these together

2) Use more than one cell to carry out more complex operations

3) Use labels to designate cell references

4) Learn to use and interpret the "formula" printout version of the Print command sequence.

Some applications require that some or all formulas be stripped from a spreadsheet, leaving the resultant values in place. The easiest way to do this as follows.

1) Note the cell reference and unit type (that is, decimal, monetary, etc.) of the cells from which the formulas are to be removed

2) Save the original spreadsheet

3) Begin a new spreadsheet

4) Enter zero (0) plus the correct unit type in each cell

5) Save the new spreadsheet

6) Load the original spreadsheet

7) Merge the new spreadsheet with the old.

You now have the original spreadsheet with the desired formulas removed, leaving just raw numbers in those cells.

Another hangup with QL Abacus users is getting printouts of spreadsheets to appear the way they want them to (or, in some cases, getting printouts at all). Format-

ting the screen presentations of spreadsheets does not appear the main problem.

The method of specifying instructions to the printer recommended by the QL User Guide and other books and articles (i.e., use of the CHR() function in the grid) has significant limitations:

1) It may cause displacement of text or data on the printout
2) It cannot be used to specify the typeface of the upper grid border
3) It is time consuming and bothersome.

There are better ways: the PREAMBLE, POSTAMBLE, and TRANSLATE sections of the QL Abacus printer driver may be modified using INSTALL_BAS to do this automatically or more simply. I recommend that you produce three printer drivers (PRINTER_DAT files): one for PICA, ELITE, and CONDENSED typeface pitch, that you include these on your working copy of the QL Abacus program cartridge as PRINTER1_DAT, PRINTER2_DAT, and PRINTER3_DAT and that you select which you will use from the QL Abacus program using the 'Delete' and 'Backup' options of the Files command sequence. First, "Delete" MDV1_PRINTER_DAT; then, 'Backup' your selection as MDV1_PRINTER_DAT using a command such as "Backup MDV1_PRINTER3_DAT as MDV1_PRINTER_DAT" from the Files command sequence.

Modify your printer drivers in the following way:

1) Reset the QL, key F1, then place a working copy of QL Abacus in Microdrive 1
2) COPY MDV1_PRINTER_DAT TO MDV1_PRINTER1_DAT (this is PICA, 10cpi, pitch)
3) Key in and enter LRUN MDV1_INSTALL_BAS
4) Select EPSON FX-80 and make a screen copy of it by keying F1
5) Give the file a new name, e.g., "print3cond" and modify the PREAMBLE by keying in and entering the the following:

```
27,,27,'R,0,27,71,15
```

(this sets up CONDENSED, 17cpi, pitch)

6) Modify the POSTAMBLE by keying and entering

```
27,72,18        (resets PICA pitch)
```

7) Enter the following codes in TRANSLATE 2 to 5, respectively:

```
'_,27,45,1          (sets underline on)

'-,27,45,0          (sets underline off)

'~,27,69            (sets emphasis on)

"\,27,70            (sets emphasis off)
```

(this establishes symbols which produce the indicated printed result--~ is the tilde)

8) Install PRINTER_DAT by keying F5
9) Exit INSTALL_BAS and return to SuperBASIC mode
10) COPY MDV1_PRINTER_DAT TO MDV1_PRINTER3_DAT
11) Reset the QL and repeat the procedure from step 3, if you desire to set up a printer driver for ELITE type. (PREAMBLE code 27,,27,'R,0,27,66,2, POSTAMBLE 18)

```
    A       B       C       D       E       F       G       H       I       J
 1
 2                                  Solutions, Inc.
 3
 4                          SEMI-ANNUAL TRANSPORTATION COSTS 1/1987
 5
 6
 7
 8  SolTC187    January February   March   April     May     June    Totals
 9
10  Leasing        3400     3600    3400    3400    3200    3200     20400
11  Labor          2400     2400    2400    2700    2700    2700     15300
12  Fuel           9240     9800    9750   10200   10600    9700     59490
13  Overhead        650      687     690     660     690     690      4057
14
15  Total Costs   15890    16487   16440   16960   17190   16290     99247
16
17  Miles         72600    75000   76000   77500   79000   74900    455000
18  Gallons        6600     7000    7100    6800    7100    6500     41100
19  MPG           11.00    10.71   10.70   11.40   11.13   11.52     11.08
20  MPDOLLAR       4.57     4.55    4.62    4.57    4.60    4.60      4.58
21
22  BOTTOM LINE:          BUDGET $102,440   COSTS $99,247   SURPLUS  $3,193
23
24  05/10/1987
```

Figure 1

Figure 1 is an actual-size printout of a QL Abacus spreadsheet, including grid borders, in condensed, double-struck typeface (17 characters-per-inch). Note that the top line is in expanded typeface and the title underlined. The only way to specify the typeface pitch of the upper grid border is by modifying the PREAMBLE of the printer driver using INSTALL_BAS.

NEW QL QUILL PRODUCTS: QL TURBOQUILL

The QL software programs are now orphans. Psion Ltd. does not intend to upgrade them. But we have seen many upgrade programs that are "patched" with QL Quill or multitasked with it to make it better. KEYDEFINE, QSPELL, Q_CALC, CAPS LOCK indicators, etc., are some examples. But the most important QL Quill accessory program to date is Athene Consultants' QL TURBOQUILL. This subtle "patch" to QL Quill reduces the "mass" of the cursor--only a white underscore and overscore mark _ remain--permitting--wonder of wonders--a significant speed-up of many screen operations. Copy and Erase operations, and all other cursor movements, are made much more rapid. QL TURBOQUILL works with KEYDEFINE and TASKMASER, and QL TURBOQUILL+ (with CAPS LOCK indicator and other enhancements will soon be available-I'll let you know when). Order QL TurboQuill from Athene Consultants, 33 Holly Grove, Fareham Hants. PO16 7UP, England, UK (Tel. [0329] 282083) for #11.20 (about $19). Specify your version of Quill when ordering.

FINAL TIP

Did you know that you can speed-up QL Quill and other QLsoftware program operation significantly by merely keying F2 and removing the prompt section at the top of the screen? Try it, you'll like it.

NEXT TIME: QL Abacus "Tips," Part II, "Everything you ever wanted to know about about QL Abacus Functions and Formulas" plus more new QL products inFormation.

RESOURCES FOR YOUR SINCLAIR
FROM TIME DESIGNS MAGAZINE COMPANY

TIME DESIGNS MAGAZINE

AMERICA'S LARGEST TIMEX SINCLAIR MAGAZINE

ONLY $15 for one year
(six bi-monthly issues)

GREAT BOOKS FOR YOUR SINCLAIR

OFFICIAL 2068 TECHNICAL MANUAL.....$25.00

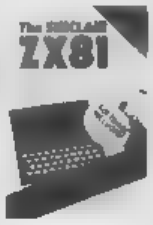
CONTROL THINGS with your Timex/Sinclair by R.L. Swarts.....$5.00

ZX81 Programming for Real Applications by Randle Hurley.....$6.00
Includes 12 Software Programs on a Cassette Tape

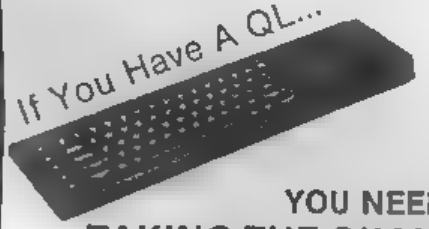
Build a Microcomputer - Controlled Robot by B.C. Taylor.....$17.00

The Best of TIME DESIGNS Volume 1.....$10.00

Taking the Quantum Leap by Mike de Sosa.....$26.00



BOTH "Control Things" and "ZX81" Package For Only **$10.00**



NEW QL BOOK!

YOU NEED
TAKING THE QUANTUM LEAP:
The Last Word on the Sinclair QL
by
Mike de Sosa

The only QL book to cover the fastest advances in hardware and software. Chock full of useful programs and examples. Fully illustrated, 280 pages. No QL owner should be without it!

$26.00

TIME DESIGNS MAGAZINE BACK ISSUES
$3.00 EACH

Nov/Dec 85 Vol 2
Jan/Feb 86
March/April 86
May/June 86
July/August 86
Sept/Oct 86
Nov/Dec 86 Vol. 3
Jan/Feb 87
March/April 87

Volume One issues are no longer available individually (See "BEST OF TDM").

The Best of TIME DESIGNS Volume 1
$10.00

Volume 2 Package

SAVE $1.00 when you order all six Volume Two issues!!
$17.00

QTY	ITEM		AMOUNT
	TIME DESIGNS MAGAZINE Subscription One Year	$15	
	TIME DESIGNS BACK ISSUES Specify Which	$3.00 each	
	THE BEST OF TDM Vol. 1	$12	
	Volume 2 Package	$17	
	TS 2068 Technical Manual	$25	
	Control Things With Your Timex Sinclair	$5	
	ZX81 Programming for Real Applications (Includes tape)	$6	
	SPECIAL PACKAGE DEAL. Both "Control Things" and ZX81"	$10	
	Build a Microcomputer Controlled Robot	$17	
	Taking The Quantum Leap	$26	

WE PAY THE POSTAGE ON ALL ORDERS! **TOTAL**

FOR MAGAZINE SUBSCRIPTION (please circle one below)
New Subscription This is a Renewal This is a Gift

SHIP TO:
NAME
ADDRESS
CITY/STATE
ZIP
PHONE
PLEASE PRINT NEATLY. SEND ORDER TO ADDRESS BELOW.

PAYMENT
Check or M/O.
CREDIT CARD
MASTERCARD / VISA
Acct. Number
Exp. Date

24 hour ORDER LINE
(503) 824 2658
For Books, Subscriptions and Renewals

VISA and MASTERCARD ACCEPTED

TIME DESIGNS MAGAZINE CO.
29722 Hult Rd., Colton, OR 97017

the CLASSIFIED

FREE ADS FOR SUBSCRIBERS

FOR SALE: 2040 Printer at $25; and 20 Timex cassettes plus 2 others at $25 for all. Like new, all with instructions/original boxes. 207 Fairway Dr., Mechanicsburg, PA 17055 or (717) 766-0451.

FOR SALE: TS1000 in metal cabinet with keyboard, Westridge modem, Aerco C.P. Interface, Memotech 64K RAM. Send for complete hardware & software list to: Richard Beier, 1 Darwin Dr., N. Merrick, NY 11566.

FOR SALE: BYTE BACK Centronics Interface for the 2068. $70 ppd. UPS. Ted Dupar, 21696 Cabrosa, Mission Viejo, California, 92691.

FOR SALE: TS2068 Computer- $75, Sakata SC-100 Monitor- $200. Many other items (2040 Printer, Rotronics Wafadrive, Rainbow I/F, software on wafer & cassette, computer desk) All items sold "as is". Write for complete list: John Solonsky Jr, 2410 Prinston, Alamogordo, NM, 88310 or (505) 437-8636.

SALE: TS1000 (no TV switch) $17; 9v wall adapter $6; 16K expander $20; 2040 Printer $40; manuals, MC texts, books, 2068 programs, etc., for info- SASE to: F. Dupre, 10 Charlotte Ave., Southhampton, PA 18966.

SINCUS NEWS- $8/yr=6 issues. TS2068 oriented user group newsletter. Mail check to SINCUS, 1229 Rhodes Rd., Johnson City, NY 13790; a non-profit amateur users-oriented group.

SOFTWARE FOR THE 2068. Send SASE for FREE catalog & price list to: E. Ray Rash, 2424 S.W. 78th St., Oklahoma City, OK 73159.

WANTED: 2068 hardware - disk drive or A&J, modem, interfaces. Software - graphics, tape to disk conveter, modem s/w, games. Send your price list. L Biederman, P.O. Box 5068, Ft McClellan, AL 36205.

FOR SALE: Spring cleaning has produced too many items in little space, forcing to sell. Many new and slightly used items for Timex Sinclair owners. Hardware and software. Send SASE for complete list and prices. Tim Woods, c/o TIME DESIGNS, 29722 Hult Rd., Colton, OR 97017.

FOR SALE: TSUG of Las Vegas has new, unopened TS1000 and TS2068 software. Send SASE for list and prices to: Ralph Hammer, 7 Baer Dr., Las Vegas, NV 89115.

TRADE: A&J (2068) Microdrive with wafers, for TS1000 peripherals, "unusual" software, or other "what have you's". Make an offer. Fred Henn, c/o Rockelman & Henn Pump Co., Inc., 1333 Military Rd., Buffalo, NY 14217.

SPDOS for 40 or 80 TPI Oliger systems, now available $24.95 + $1.50 for postage. Write: Cuyahoga Valley Software Works, 615 School Ave., Cuyahoga Falls, OH 44221.

NEED INFO on parallel interface for TS2068 & Canon S16 Typewriter. Can anyone out there help? Elmer Swicord Jr, RR#9, Box 339, Gulfport, MS 39503, or (601) 896-1307.

WANTED: TS-2050 Modem Manual and a 2050 Modem case. Dane Stegman, 26 Marshall Ave, Akron, NY 14001.

ZX80 (both ROMs); TS1000; 16, 32, 64K RAMS; 2 keyboards; parallel I/F; TRS80 recorder; ZX Printer (15 rolls of paper); battery b.u., MemoCalc; MemoText; cables; software-books-mags-manuals-etc. $150. Kristjan Alfjordsson, 1409 Kuehner Dr. #161k, Simi Valley, CA 93063. SASE for info.

Any Club Librarian interested in swapping Public Domain program tapes should contact Algis Georis, 355 Royal Oak Blvd., Richmond Hts., OH 44143, (216) 481-8205.

MONEY MACHINE $12, Diamond Mike $15. 2068 software on Aerco disk or tape. New/used hard/soft-ware for 1000 & 2068. 22-cent stamp for catalog. Chia-Chi Chao, 73 Sullivan Dr., Moraga, CA 94556.

NOW!!! Affordable 4-color HI-RES plotting for your 2068 and the Commodore 1520 printer/plotter! The 1520 is selling for $49.95 & the I/F hard/software is avail. from John McMichael, 1710 Palmer Dr., Laramie, WY 82070. Send SASE for complete info & sample plot.

FOR TRADE: I would like to trade my 3 1/2" SPDOS (Ramex) U1-2 Master Disk No.T28521 for a 5 1/4" version of the same, also will trade Z-TERM 64 for MTERM II tape. Telephone (218) 346-2129. After 6pm.

WANTED: 2068 Command Cartridges (must be in working order with instructions). Send titles & prices to P. Bingham, POB 2034, Mesa, AZ 85204.

FOR SALE: Tasman RS232 Serial interface for TS2068. $30.00 includes First Class postage and insurance. Darrell K Strong, 1043 Kinau St., #404, Honolulu, HI 96814

FOR SALE: TS2068 with A&J Microdrive, Gorilla Banana printer, many extras, $150. Guy Nicholson, 40863 Laredo, Cherry Valley, CA 92223. (714) 845-4913.

Continued Next Page.

Do you have some equipment or a program that you would like to sell? Looking for something hard to find? Place an ad in **THE CLASSIFIEDS!** Subscribers can place one free personal ad in each issue. Ad size is 32 Col. wide (like 2040 paper) and maximum of six lines. For additional lines - $3 each. **NON-SUBSCRIBERS** and **DEALERS**: $4 a line. **DEADLINE FOR ALL CLASSIFIED ADS:** Two weeks before publication date. Mail your ad to: **TIME DESIGNS MAGAZINE, The Classifieds Dept., 29722 Hult Rd., Colton, Oregon 97017.**

CLASSIFIED

MILLENNIA K USERS: New Utilities- If Then Else structure checks for disk fn, RENUM works on designated block of BASIC, NAMES and DUMP reveal what's in VARS, MAKE and delete REM's; PLUS USR file CHECKER, and SECTOR EDITOR. Utilities work from BASIC-like SPDOS calls. Full documentation. Only $11 for (DSQD), $12 for cassette ppd. COMING SOON: RESTORE to unERASE files, ALPHA to put order in your CAT. Check or m/o to: Munson Cockayne, 342 Trotter Ct., Sanford, FL 32771.

FOR SALE: TS2068 and Aerco printer interface; Flight Simulator & many other cassettes...$100. TS1000, modem, printer, terminal I/F for HAM Rtty, CW, Wepax...for $100. Call: Donn KA0SOH (314) 832-8617 or write: 5814 Minnesota, St. Louis, MO 63111.

FOR SALE: Alphacom 32 Printer, w/ interface & power supply $30. Textwriter 64, Artworx V1.1, Zeus Assembler, Ckt. Bd. Scramble (all for 2068) $40 for all, or trade for full Aerco W/P. W. Flower, 1804 Vincennes St., #22, Northridge, CA 91325.

NEW SOFTWARE for the TS1000 - KAPKIT 1000. KK is a great utility program. Send $14.95 + $1.95 S&H or SASE for more information. Chuck Peterson, PO Box 62, Alcester, SD 57001.

FOR SALE: TS1000 programs, TS2068, TS2068 Tech Manual. For list and prices, send large SASE to D.H. Berry, 2106 Opal Dr., Orlando, FL 32822.

Subscribe to TIMELINE2/SincLink! Get User Group discounts on A&J Microdrive products. Send $10 to: SincLink, 6675 Clifford Dr., Cupertino, CA 95014 or (408) 253-3175. SASE for price list.

NEW PRODUCTS ANNOUNCEMENT
For TS2068 and SPECTRUM

NovelSoft

brings Artificial Intelligence
to the Timex/Sinclair

Introducing...

ZXPERT

ZXpert lets you explore the reasoning process of human experts.

With ZXpert you can create an Expert System on any topic you like!

- **A useful and educational program**
- **Easy and fun to use**
- **No programming language to learn**
- **On-line help**
- **Includes 2068 & Spectrum versions**
- **Supports Microdrives**
- **Two example knowledge bases**
- **Superb Manual with Tutorial**

plus...

Once you have developed a useful knowledge base, with your permission NovelSoft will sell it and pay you a royalty!!!

Join the revolution with...ZXpert

NovelSoft
A FORMAT FOR THE FUTURE

33 Skelton Street, Toronto, Ontario, Canada M8V 3W3 * TEL. (416)259-8682 * CompuServe 70416,1435

WEYMIL presents...

A small collection of truly innovative products for Sinclair computers

DOES YOUR TS2068 NEED A DISK DRIVE?

LET US INTRODUCE THE LARKEN DISK SYSTEM

The LARKEN DISK SYSTEM represents both the most flexible and most cost effective method of adding a disk drive to your TS2068. The system consists of the LKDOS-EXTENDED BASIC CARTRIDGE and the LATEST 2068/SPECTRUM DISK INTERFACE. The cartridge has the complete operating system (LKDOS) and 10 Extended Basic Commands for Utilities, Scrolling Windows, and Graphics. It also uses all of the keyboard commands such as FORMAT ERASE PRINT LOAD MERGE and more. Since the LKDOS is cartridge-based, your *entire* system memory is at your disposal.

The LATEST 2068/SPECTRUM DISK INTERFACE will handle up to FOUR drives. You can store up to 400K on a standard double-density, double-sided floppy. That's 40K more than an IBM PC. It also supports single-sided and the new quad density drives as well. You have the option of choosing either 5¼" or the newer 3" drives. The interface includes a built-in Kempston Joystick port and a handy "snapshot" NMI pushbutton feature to save screens at any point. This is an especially powerful feature for graphics applications.

The LARKEN DISK SYSTEM is designed to give you full disk drive power without draining your budget.

LARKEN DISK SYSTEM for the TS 2068 only	$ 110.00
shipping and handling	5.00

EXPLORE DESKTOP PUBLISHING! STAN LEMKE'S PIXEL PRINT

Stan Lemke has won critical acclaim as one of N.America's finest software authors. His products have become standard items in many software libraries. Stan has worked hard to bring the rapidly growing world of desktop publishing to the TS 2068 user.

PIXEL PRINT will allow you to import text as well as graphics. It also has many fonts of its own for you to use. You can import graphics from most graphics software such as PIXEL SKETCH & GRAPHICS EDITOR or ARTWORX. There is even a facility to allow you to import graphics from Zebra's unique Greeting Card Designer.

PIXEL PRINT gives you very professional results without the drudgery of messy manual cut-and-paste operations. You can format PIXEL PRINT for a wide variety of printers including STAR, GEMINI, and C. ITOH in addition to EPSON compatibles. Text and graphics can easily be moved with one-pixel accuracy for precise placement.

If you produce a newsletter magazine, posters, or whatever, PIXEL PRINT will speed your task. You can print either a 4" by 10½" column or 8½" by 11" full page. Even a novice can easily combine the features of PIXEL PRINT to produce superior results. Fully documented. Let your imagination soar with Stan Lemke's latest product.

PIXEL PRINT for the TS 2068 only $20.00

THE GRAPHICS CHAMPS!

THE BEST SELLING GRAPHICS PACKAGES FOR THE TS 2068

PIXEL SKETCH & GRAPHICS EDITOR V20 is Stan Lemke's original graphics package. It has had excellent user group reviews and is a snap to use. Well written, step-by-step documentation guides you effortlessly from loading to producing your own "masterpiece." Great pixel and text placement control.

PIXEL SKETCH & GRAPHICS EDITOR V20 only $19.94

ARTWORX V1.1 establishes a new standard for color graphics with features normally not found for small systems. Auto speed control, pull down menus, cut and paste windows, ZOOM, and more. Everything except text entry is joystick controlled. Excellent documentation. So easy to use that many users never read the instructions.

ARTWORX V1.1 only $20.00

SPECIAL DESKTOP PUBLISHING PACKAGE OFFER!

Order both PIXEL PRINT and PIXEL SKETCH & GRAPHICS EDITOR

for only $ 34.95 !

SAVE $ 5.00 !

SPECIAL OFFER!!!

Save $5.00 when you order the combination of THRUST, RIGTER JOYSTICK INTERFACE, and KRUNCHER 1000 and pay only $ 64.95

SHIPPING INSTRUCTIONS: Please enclose $2.00 shipping and handling with your order.

WEYMIL CORPORATION

Box 5904

BELLINGHAM, WA 98227-5904